
Flat Fingerprint Classification Using a Rule-Based Technique, Based on Directional Patterns and Singular Points



Author:

Kribashnee Dorasamy

Co-supervisor:

Leandra Webb-Ray

Supervisor:

Prof. Jules Tapamo

*A dissertation submitted in fulfilment of the requirements
for the degree of Master of Science in Computer Engineering*

College of Agriculture, Engineering and Science, University of KwaZulu-Natal

May 2016

Preface

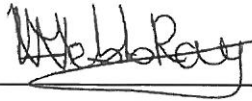
The research discussed in this dissertation was carried out in the College of Agriculture, Engineering and Science of the University of KwaZulu-Natal, Durban from July 2014 until January 2016 by Ms. Kribashnee Dorasamy (208504643) under the supervision of Prof. Jules-Raymond Tapamo and co-supervision of Ms. Leandra Webb-Ray. As the candidate's supervisor I, Prof. Jules-Raymond Tapamo agree/do not agree to the submission of this thesis.

Signed: _____

Date: _____

As the candidate's co-supervisor I, Ms. Leandra Webb-Ray agree/~~do not agree~~ to the submission of this thesis.

Signed: _____



Date: _____

18 May 2016

I, Ms. Kribashnee Dorasamy (208504643), hereby declare that all the materials incorporated in this dissertation are my own original work, except where acknowledgement is made by name or in the form of a reference. The work contained herein has not been submitted in any form for any degree or diploma to any institution.

Signed: _____



Date: 22 / 06 / 2016

Declaration 1- Plagiarism

I, Kribashnee Dorasamy, declare that,

1. The research reported in this dissertation, except where otherwise indicated, is my original research.
2. This dissertation has not been submitted for any degree or examination at any other university.
3. This dissertation does not contain another person's data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.
4. This dissertation does not contain another person's writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:
 - (a) Their words have been re-written but the general information attributed to them has been referenced
 - (b) Where their exact words have been used, then their writing has been placed in italics and inside quotation marks, and referenced.
5. This dissertation does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the dissertation and in the References section.

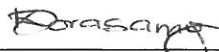
Signed: Kribashnee Dorasamy

Date: 2016 / 06 / 22

Declaration 2- Publications

I, Kribashnee Dorasamy, declare that the following publications came out of this dissertation

1. K. Dorasamy, L. Webb, J. Tapamo, and N. Khanyile, "Fingerprint Classification using a Simplified Rule-Set based on Directional Patterns and Singularity Features", in *8th IAPR Int. Conf. of Biometrics.*, IEEE, Phuket, Thailand, 2015, pp. 400–407.;
2. K. Dorasamy, L. Webb, and J. Tapamo, "Evaluation of the Changes in Directional Patterns Under the Variation of Rotation and Number of Regions", in *the Int. Conf. of the Biometrics Special Group (BIOSIG).*, IEEE, Germany, 2015, pp. 1-8.
3. K. Dorasamy, L. Webb-Ray, and J. Tapamo, "Optimal Directional Patterns for Complete and Incomplete Fingerprint Classes", *Elsevier Image and Vision Computing* (under review).
4. K. Dorasamy, L. Webb-Ray, and J. Tapamo, "A Missing Singular Point Resistant Fingerprint Classification Technique, Based on Directional Patterns", *Elsevier Expert Systems with Applications* (under review).

Signed: 

Date: 22 / 06 / 2016

Abstract

College of Agriculture, Engineering and Science, University of KwaZulu-Natal

Master of Science in Computer Engineering

Flat Fingerprint Classification Using a Rule-Based Technique, Based on Directional Patterns and Singular Points

by Kribashnee Dorasamy

The demand to keep up with current technology trends has lead to the development of compact scanner devices. Owing to these hardware constraints, fingerprints are often not fully captured and crucial fingerprint landmarks, known as Singular Points (*SPs*) are excluded. The regions around these points hold key characteristics to determine a fingerprint class. The loss of *SPs* make it difficult to identify class patterns which directly impacts the fingerprint classification stage of the recognition system. Fingerprint classification is important since it reduces the search time by only comparing an input fingerprint to fingerprints in the database (*DB*) of the same class pattern, to find a possible fingerprint match. When misclassification occurs, it takes longer to find a match as additional classes must be searched.

Using the correct features to identify the class pattern may provide the capabilities to handle lack of information and prevent misclassification. From the analysis conducted in this dissertation, Directional Patterns (*DPs*) were found to be a promising feature to overcome missing *SPs* due to their global representation of the fingerprint. Furthermore, investigations showed that detecting and aligning the remaining *SPs* in a specific way, allowed unique and consistent class patterns to appear. The alignment highlighted unique structural characteristics of each flat fingerprint case. Due to this more robust and simpler classification rules could be developed.

In this dissertation, classification rules for flat fingerprint cases including Whorl (*W*) with two loops/deltas; Right Loop (*RL*), Left Loop (*LL*), Tented Arch (*TA*) and *W* with single top loop and single delta; *RL*, *LL* and *W* with single loop; *RL* and *LL* with a single delta; and lastly Plain Arch (*PA*) and Partial Fingerprint (*PF*) cases were implemented. An overall accuracy of 92.48% was obtained on the Fingerprint Verification Competition (*FVC*) 2002 *DB1* and 2004 *DB1*, illustrating that *DPs* combined with the available *SPs* are capable of classifying flat fingerprint successfully.

Acknowledgements

I would like to thank my supervisor Prof. Jules-Raymond Tapamo and my co-supervisor Ms. Leandra Webb-Ray for their support and insight. Their knowledge has contributed greatly towards my research. Secondly, I would like to thank Ms. Portia Nontokozo Khanyile, Mr. Yaseen Molla and Mr. Luke Darlow for assisting me with proof reading. Lastly, I would like to thank my friends and family for encouraging me to achieve all my dreams.

Contents

Preface	i
Declaration 1- Plagiarism	ii
Declaration 2- Publications	iii
Acknowledgements	v
Abstract	iv
Contents	vi
List of Figures	ix
List of Tables	xiii
Abbreviations	xiv
1 Introduction	1
1.1 The history of fingerprint authentication	1
1.2 Fingerprint classification	2
1.3 Moving towards fingerprint automated systems	3
1.3.1 Brief description of fingerprint global features	4
1.4 Emergence of live fingerprint scanners	5
1.5 Motivation	6
1.6 Problem statement	7
1.7 Research question	8
1.8 Research goal	8
1.9 Research objectives	8
1.10 Delineation	9
1.11 Contributions	9
1.12 Brief chapter overview	10
2 Literature Review	11
2.1 Introduction	11
2.2 Selecting the optimal features to overcome missing SPs	13

2.2.1	Advantages and disadvantages of fingerprint features	13
2.2.2	Orientation feature	15
2.2.3	Benefits of combining SPs with orientation features	16
2.3	Fingerprint classification method designed to overcome the challenges of missing SPs	18
2.4	Limitation of local orientation fields	20
2.5	Representation of classes using global orientation fields	21
2.6	Directional pattern approach	23
2.7	Conclusion	24
3	Evaluating the Change of Fingerprint Directional Patterns Under Rotation and Number of Regions	26
3.1	Introduction	26
3.2	Construction of a DP with n regions	29
3.2.1	Pre-processing fingerprints	29
3.2.2	Orientation Field Algorithm	29
3.2.2.1	Normalisation	30
3.2.2.2	Ridge orientation estimation	30
3.2.3	Region segmentation for n regions	32
3.3	Experimental set-up	33
3.3.1	Experiment 1: Evaluating the impact of varying the number of regions	33
3.3.1.1	Singular point detection	34
3.3.1.2	Observation made on the DP	35
3.3.2	Experiment 2: Evaluating the effect of rotation on DPs	35
3.3.2.1	Increments for rotation	35
3.3.2.2	Observation made on the DP	35
3.3.3	Testing Samples	36
3.4	Discussion	36
3.4.1	Observations made on the DPs with different the number of regions	36
3.4.1.1	Impact of the varying the number of regions on the SP detection accuracy	36
3.4.1.2	Impact of varying the number of regions on the amount of visible noise present on DPs	39
3.4.2	Variation of rotation	40
3.4.2.1	Observation of the W DP with two loops and two deltas, under rotation	40
3.4.2.2	Observation of the W, RL, LL and TA DPs with one loop and one delta, under rotation	41
3.4.2.3	Observation of RL, LL, TA and W DPs with one loop under rotation	47
3.4.2.4	Observation of the RL and LL DPs with one delta, under rotation	49
3.4.2.5	Observation of the PA and PF DP with no SPs under rotation	51
3.5	Recommendation	54
3.6	Conclusion	54

4	A Missing Singular Point Resistant Fingerprint Classification Technique	56
4.1	Introduction	56
4.2	Classification Set-up	58
4.2.1	Pre-processing and orientation field estimation	59
4.2.2	Region segmentation	59
4.2.3	SP detection	60
4.3	Implementation of classification rules	60
4.3.1	Flat fingerprint case containing a loop and a delta	61
4.3.1.1	Detecting a bottom loop	63
4.3.1.2	Vertically aligning SPs for a fingerprint with a single loop and single delta	66
4.3.1.3	Rule for DPs with two CRs	67
4.3.1.4	Rules for DPs with three CRs	68
4.3.2	Flat fingerprint case with a single loop	69
4.3.2.1	Rotation of fingerprints with a single loop	69
4.3.2.2	Rules to classify a DP with a single loop	71
4.3.3	Rule for flat fingerprint cases with a single delta	73
4.3.3.1	Rotation of fingerprints with a single delta	73
4.3.3.2	Rule for DPs with a single delta	75
4.3.4	Classifying DPs with no SPs	76
4.3.4.1	Alignment of fingerprints containing no SPs	77
4.3.4.2	Rule for a DP with no SPs	80
4.4	Testing	83
4.4.1	Experimental context	83
4.4.2	Test data	83
4.4.3	Measurements	84
4.4.4	Software used for experimental testing	84
4.5	Conclusion	84
5	Experimental Results	86
5.1	Introduction	86
5.2	Results	87
5.3	Conclusion	94
6	Conclusion	95
6.1	Summary	95
6.2	Future work	96
	References	98

List of Figures

1.1	The five most common classes namely, (a) <i>RL</i> , (b) <i>LL</i> , (c) <i>TA</i> , (d) <i>PA</i> , and (e) <i>W</i>	3
1.2	Singular Points (<i>SPs</i>):(a) loop and (b) delta	4
1.3	The success of tracking fingerprints using <i>AFIS</i> during 1986 to 1988 [1]	6
1.4	Small inter-class variability issues between class (a) <i>RL</i> (b) <i>LL</i> and (c) <i>TA</i>	6
2.1	<i>W</i> fingerprint represented by block-wise orientation fields[2]	15
2.2	An illustration of a (a) <i>RL</i> and it's (b) global pattern (structural representation) with four homogeneous regions [3]	21
2.3	Structural representation of <i>LL</i> fingerprints using the greedy method which produces (a) three regions and (b) six regions [4]	22
2.4	Example of a (a) <i>TA</i> and it's (b) <i>DP</i> [5]	23
2.5	Eighteen pattern zones with deltas and loops [5]	23
3.1	Example of (a) a four region <i>RL DP</i> and, (b) a nine region <i>RL DP</i> [6]	27
3.2	Examples showing, (a) an upright <i>RL</i> , (b) a rotated <i>RL</i> , (c) <i>DP</i> of an upright <i>RL</i> and (d) <i>DP</i> of a rotated <i>RL</i> [5]	27
3.3	The axis used to calculate the orientation fields	31
3.4	A <i>DP</i> created using three orientation ranges [7, 8]	32
3.5	Flow diagram of the experimental set-up used to evaluate the impact of varying the number of regions, on the <i>SP</i> detection accuracy and the amount of visible noise produced [8]	33
3.6	An example illustrating a <i>SP</i> where all three region intersect using a 24 <i>ND</i> [7] .	34
3.7	Type of <i>SPs</i> (a) loop and (b) delta [7]	34
3.8	Flow diagram of the experimental set-up that evaluates the effect of varying the angle of rotation	35
3.9	<i>SPs</i> detection versus number of regions (<i>n</i>) by varying the value of the neighbourhood (<i>ND</i>) and the pixel tolerance (<i>PT</i>) [8]	36
3.10	A <i>W</i> class represented by a (a) 15 region <i>DP</i> , and (b) zoomed in at its converging region, which is not a <i>SP</i> [8]	38
3.11	(a) Noisy fingerprint image with <i>DPs</i> formed using different numbers of regions namely: (b) 3, (c) 4, (d) 5, (e)7, (f) 12 and (g) 15 [8]	39
3.12	The <i>DPs</i> of an upright <i>W</i> that is rotated at angle, (b) 0°, (c) 10°, (d) 50° and (e) 90° [8]	41
3.13	Indication of a top loop and a bottom for <i>W</i> fingerprint image	41

3.14	Indication of a top loop for (a) <i>RL DP</i> , (c) <i>LL DP</i> , and both a top loop and bottom loop on (d) a <i>W DP</i>	42
3.15	Three different types of <i>DP</i> layouts produced from (a) an <i>RL</i> fingerprint that forms a 3- <i>CR</i> layout, (b) a 3- <i>CR</i> layout, (c) an <i>RL</i> fingerprint that forms a 2- <i>CR</i> layout, (d) a 2- <i>CR</i> layout, (e) an <i>RL</i> fingerprint that forms a 1- <i>CR</i> layout and (f) a 1- <i>CR</i> layout [8]	43
3.16	Three different types of <i>DP</i> layouts produced from (a) an <i>LL</i> fingerprint that forms a 3- <i>CR</i> layout, (b) a 3- <i>CR</i> layout, (c) an <i>LL</i> fingerprint that forms a 2- <i>CR</i> layout, (d) a 2- <i>CR</i> layout, (e) an <i>LL</i> fingerprint that forms a 1- <i>CR</i> layout and (f) a 1- <i>CR</i> layout [8]	43
3.17	Two different types of <i>DP</i> layouts produced from (a) a <i>TA</i> fingerprint that forms a 3- <i>CR</i> layout, (b) a 3- <i>CR</i> layout, (c) a <i>TA</i> fingerprint that forms a 2- <i>CR</i> layout and (d) a 2- <i>CR</i> layout [8]	43
3.18	The <i>DPs</i> formed when an <i>RL</i> fingerprint that produces a 3- <i>CR</i> layout at it's upright position is rotated at angles, (b) 0°, (c) 30°, (d) 80°, (e) 120°, (f) 140°, (g) 160° and (h) 180° [8]	43
3.19	The <i>DPs</i> formed when an <i>RL</i> fingerprint that produces a 2- <i>CR</i> layout at it's upright position is rotated at angles, (b) 0°, (c) 20°, (d) 50°, (e) 60°, (f) 80° and (g) 120° [8]	43
3.20	The <i>DPs</i> formed when an <i>RL</i> fingerprint that produces a 1- <i>CR</i> layout at it's upright position is rotated at angles, (b) 0°, (c) 10°, (d) 20°, (e) 30°, (f) 60°, (g) 90° and (h) 150° [8]	44
3.21	The <i>DPs</i> formed when an <i>LL</i> fingerprint that produces a 3- <i>CR</i> layout at it's upright position is rotated at angles, (b) 0°, (c) 10°, (d) 60°, (e) 110°, (f) 160°, (g) 210°, and (h) 280° [8]	44
3.22	The <i>DPs</i> formed when an <i>LL</i> fingerprint that produces a 2- <i>CR</i> layout at it's upright position is rotated at angles, (b) 0°, (c) 90°, (d) 130°, (e) 220°, (f) 260°, (g) 300° and (h) 340° [8]	44
3.23	The <i>DPs</i> formed when an <i>LL</i> fingerprint that produces a 1- <i>CR</i> layout at it's upright position is rotated at angles, (b) 10°, (c) 60°, (d) 90°, (e) 130°, (f) 260°, and (g) 300° [8]	44
3.24	The <i>DPs</i> formed when a <i>TA</i> fingerprint that produces a 3- <i>CR</i> layout at it's upright position is rotated at angles, (b) 10°, (c) 50°, (d) 90°, (e) 190°, (f) 250° and (g) 350° [8]	44
3.25	3- <i>CR</i> layouts at an angle of 0° that was formed by classes (a) <i>RL</i> , (b) <i>LL</i> , and (c) <i>TA</i> [8]	46
3.26	2- <i>CR</i> layouts at an angle of 10° that was formed by classes (a) <i>RL</i> , (b) <i>LL</i> , and (c) <i>TA</i> [8]	46
3.27	1- <i>CR</i> layouts at an angle of 0° that was formed by classes (a) <i>RL</i> , (b) <i>LL</i> , and (c) <i>W</i> [8]	46
3.28	Fingerprints which are rotated such that the <i>SPs</i> are vertically aligned produces a 2- <i>CR</i> layout of class, (a) <i>LL</i> , (b) <i>RL</i> , and (c) <i>TA</i> ; and a 3- <i>CR</i> layout of class (d) <i>LL</i> , (e) <i>RL</i> , and (f) <i>TA</i> [8]	46
3.29	The <i>DPs</i> formed from an <i>RL</i> with a single loop that is rotated at angles of, (b) 0°, (c) 50°, (d) 100°, (e) 170°, (f) 220°, (g) 310° and (h) 340° [8]	47

3.30	The <i>DPs</i> formed from an <i>LL</i> with a single loop that is rotated at angles of, (b) 0°, (c) 10°, (d) 100°, (e) 170°, (f) 220°, (g) 310° and (h) 340° [8]	47
3.31	The <i>DPs</i> formed from a <i>TA</i> with a single loop that is rotated at angles of (b) 0°, (c) 90°, (d) 130°, (e) 260°, and (f) 340°	48
3.32	The <i>DPs</i> formed from a <i>W</i> with a single loop that is rotated at angles of, (b) 0°, (c) 10°, (d) 20°, (e) 40°, (f) 90°, (g) 190° and (h) 230° [8]	48
3.33	Region 2 that links to the loop <i>L</i> found on (a) a <i>TA</i> does not extend to the side of the <i>DP</i> , whereas for (b) an <i>RL</i> it extends to the sides of the fingerprint	48
3.34	The <i>DPs</i> which are produced by classes that have a single loop, namely: (a) <i>RL</i> , (b) <i>LL</i> and (c) <i>W</i> [8]	48
3.35	The flow of the orientation fields at the bottom of the fingerprint differ for class (a) <i>RL</i> , (b) <i>LL</i> and (c) <i>W</i> when the loop direction of each of these classes points downwards	49
3.36	The different patterns formed from an <i>RL</i> with a single delta that is rotated at angles of, (b) 0°, (c) 30°, (d) 40°, (e) 80°, (f) 120°, (g) 200°, (h) 260°, (i) 330°, and (j) 350°	50
3.37	The different patterns formed from an <i>LL</i> with a single delta that is rotated at angles of, (b) 0°, (c) 30°, (d) 40°, (e) 80°, (f) 120°, (g) 200°, (h) 260°, (i) 330°, and (j) 350°	50
3.38	The different patterns formed from a rotated <i>RL</i> with a single delta that has the same length on either side of the delta. It is rotated at angles of, (b) 0°, (c) 30°, (d) 40°, (e) 80°, (f) 120°, (g) 200°, (h) 260°, (i) 330°, and (j) 350°	50
3.39	Class (a) <i>LL</i> fingerprint with a single delta, that produces (b) a unique <i>LL DP</i> , and class (c) <i>RL</i> fingerprint with a single delta, that produces (d) a unique <i>RL DP</i>	51
3.40	The different patterns formed from a rotated <i>PA</i> at angles of, (b) 0°, (c) 20°, (d) 30°, (e) 40°, (f) 60°, (g) 90°, (h) 120°, (i) 150°, (j) 210°, (k) 270°, (l) 300°, and (m) 320°	52
3.41	The different patterns formed from a rotated <i>PF</i> at angles of, (b) 0°, (c) 20°, (d) 30°, (e) 40°, (f) 60°, (g) 90°, (h) 120°, (i) 150°, (j) 210°, (k) 270°, (l) 300°, and (m) 320°	52
3.42	Unique <i>DP</i> for (a) a <i>PA</i> and (b) a <i>PF</i>	52
3.43	Highest curvature point within the circle for (a) <i>PA</i> and (b) <i>PF</i>	53
3.44	Width w_c is the smallest width on the innermost region and width w_e is the maximum width below the converging area on (a) a <i>PA</i> and (b) a <i>PF</i>	53
4.1	Brief overview of the complete classification process of the proposed algorithm	58
4.2	An example of a <i>DP</i> with three different regions each with its own orientation range [7]	60
4.3	Intersection of 3 regions that represents a <i>SP</i> using a 24 <i>ND</i> [7]	61
4.4	Decision tree of the overview of the proposed fingerprint classification algorithm, with output <i>W</i> and sub-algorithms indicated by 1, 2, 3 and 4	63
4.5	Decision tree of fingerprint classification specific to a one loop one delta rule with class outputs, <i>RL</i> , <i>LL</i> , <i>TA</i> and <i>W</i>	64
4.6	A complete <i>W</i> with two loops and two deltas, indicating a top loop and a bottom loop on (a) a segmented fingerprint and (b) a <i>DP</i>	65

4.7	A <i>DP</i> illustrating a single bottom loop <i>L</i> in class (a) <i>W</i> and a single top loop <i>L</i> in class (b) <i>RL</i>	65
4.8	Symbols depicted on a single loop and single delta <i>W</i> , that are used to perform the detection of the type of loop	65
4.9	<i>DP</i> produced from a <i>LL</i> fingerprint where the loop (<i>L</i>) and delta (<i>D</i>) are vertically aligned [7]	66
4.10	Two <i>CR</i> layouts formed from classes (a) <i>LL</i> , (b) <i>RL</i> and (c) <i>TA</i> [7]	67
4.11	The maximum width w_f , w_1 and w_3 for a <i>TA</i> [7]	68
4.12	3- <i>CR</i> layout produced by classes, (a) <i>TA</i> , (b) <i>RL</i> and (c) <i>LL</i> [7]	68
4.13	Decision tree of fingerprint classification algorithm for a single loop with class outputs, <i>W</i> , <i>RL</i> , <i>LL</i> , and <i>TA</i>	70
4.14	Loop direction pointing (a) 135° and (b) 90° [7]	70
4.15	<i>DP</i> containing a single loop (a) <i>LL</i> , (b) <i>RL</i> and (c) <i>W</i> [7]	71
4.16	A single loop <i>DP</i> that has no regions below region 2 [7]	72
4.17	<i>DP</i> of class (a) <i>RL</i> with $maxR2$ longer in length and (b) <i>LL</i> with $maxR1$ longer in length	72
4.18	Decision tree of fingerprint classification algorithm for a single delta with class outputs <i>RL</i> and <i>LL</i>	74
4.19	Fingerprint with single delta <i>D</i> where the orientation fields below <i>D</i> flows (a) appropriately 35° and (b) horizontal	74
4.20	Points of interest on a fingerprint with single delta <i>D</i>	74
4.21	A single delta <i>DP</i> illustrating the position of region 2 for, (a) a <i>LL</i> and (b) a <i>RL</i>	75
4.22	<i>DP</i> containing the symbols used to perform the classification rule of a single delta fingerprint	77
4.23	Overview of the rule used to classify a <i>PA</i> and a <i>PF</i>	78
4.24	Rotated <i>PF</i> where the orientation flow near point A is 90°	78
4.25	Points of interest on a <i>PF</i> fingerprint	79
4.26	A 10-region <i>DP</i> that represents the rotated fingerprint for (a) a <i>PA</i> and (b) a <i>PF</i>	81
4.27	A <i>PF DP</i> containing the symbols used for the algorithm	81
4.28	A <i>PF DP</i> containing row of Pt_c used for the algorithm	82
4.29	The widths used to classify a <i>PA</i> and a <i>PF</i> where (a) denotes the smallest width w_{mreg} in region R_m and (b) denotes the maximum width w_{maxi} in region R_{m2} (cropped R_m)	82
5.1	Excluded fingerprints images due to loss of <i>SPs</i> on the <i>DPs</i>	87
5.2	<i>TA</i> class that produces a <i>DP</i> that resembles the characteristics of an <i>RL</i> [1]	88
5.3	<i>RL</i> class that produces a <i>DP</i> that resembles the characteristics of a <i>TA</i>	89
5.4	Accuracy achieved from imperfect fingerprint classes with missing <i>SPs</i> from the <i>FVC 2002 DB1</i>	91
5.5	Accuracy achieved from fingerprint classes with missing <i>SPs</i> from the <i>FVC 2004 DB1</i>	92
5.6	Successful classification of a <i>PA</i> where (a) is the segmented fingerprint, (b) 3-region <i>DP</i> , (c) 10-region <i>DP</i> (d) outer region and (e) mid region	93
5.7	Successful classification of a <i>PF</i> where (a) is the segmented fingerprint, (b) 3-region <i>DP</i> , (c) 10-region <i>DP</i> and (d) outer region	93

List of Tables

2.1	Summary of the Advantages and Disadvantages of Ridge-Structure features, Frequency-based features, Orientation features, <i>SPs</i> , FingerCode and Structural features	14
2.2	Speed results for <i>PCASYS</i> model and a fingerprint classification model based on <i>SPs</i> [9]	17
2.3	All possible flat fingerprint cases including missing <i>SPs</i> and the corresponding literature that attempts to solve each case [10]	19
3.1	<i>PT</i> and <i>ND</i> values selected for testing purposes [8]	38
3.2	Summary of the observations made on the <i>DP</i> with <i>n</i> regions [8]	40
3.3	Recommendation for the method of rotation used to obtain unique <i>DPs</i> for each class of a given flat fingerprint case	55
4.1	A list of all the possible flat fingerprint cases addressed by previous literature and the proposed method	57
4.2	Calculated values that produce a three region segmentation	59
4.3	Overview of each flat fingerprint case rule that is created based on the unique property established through rotation	62
5.1	Confusion matrix of the classification using the proposed algorithm on 871 images from 880 images from the <i>FVC 2002 DB1</i>	87
5.2	Confusion matrix of the classification using the proposed algorithm on 845 images from the 880 images from the <i>FVC 2004 DB1</i>	88
5.3	Confusion matrix of the classification using the proposed algorithm on 880 images from the <i>FVC 2002 DB1</i>	89
5.4	Confusion matrix of the classification using the proposed algorithm on 880 images from the <i>FVC 2004 DB1</i>	90
5.5	Accuracy results of the proposed algorithm compared to algorithms in literature that were re-implemented	90
5.6	Accuracy results of the proposed algorithm compared to algorithms by Jung and Lee, and Guo <i>et al.</i>	90
5.7	Accuracy achieved for each fingerprint case using <i>FVC 2002 DB1</i>	93
5.8	Accuracy achieved for each fingerprint case using the <i>FVC 2004 DB1</i>	94

Abbreviations

SP	Singular Point
RL	Right Loop
LL	Left Loop
TA	Tented Arch
W	Whorl
TLW	Twin Loop Whorl
CPW	Central Pocket Whorl
AW	Accidental Whorl
PA	Plain Arch
PF	Partial Fingerprint
DP	Directional Patterns
U	Unclassifiable class
AFIS	Automated Fingerprint Identification System
PCASYS	Pattern-level Classification Automation SYSTEM
FVC	Fingerprint Verification Competition
NIST	National Institute of Standard Technology
DB	Database
SOMs	Self Organizing Maps
FBI	Federal Bureau Investigation
CR	Common Region
ND	Neighbourhood
PT	Pixel Tolerance
KL	Karhunen Loeve

PNN **Probabilistic Neural Networks**

ANN **Artificial Neural Networks**

Chapter 1

Introduction

1.1 The history of fingerprint authentication

After years of analysis and development within the biometric sector, fingerprint have become commonly used for authentication in electronic commercial devices. The Touch ID is a well known example that was introduced by Apple in 2015 [11]. Their success brought about an approximately 6 million downloads of the biometric application just within that year [12, 13]. The commercial demand to incorporate biometric authentication into smart phones, tablets and wearable devices has even predicted to drive over 4.8 billion users in 2020 [12, 13]. This demand is rapidly replacing the hassle of using PINS and passwords [13–17]. Unlike PINS and passwords, biometrics was found to be more appealing due to its reliability in differentiating between an authorised person and an imposter who tries to illegally gain access to a system [14, 18]. Consequently it increases the level of security. Even banks and leading companies like Apple and Samsung have now incorporated fingerprint authentication as a means of higher security for payment transactions [13]. Fingerprints are by far the most preferred biometric trait that is commonly incorporated into devices [13] since it is more accurate and convenient (easy to acquire), unlike other biometrics, such as face and voice [14–16, 18–21]. In addition, fingerprint biometrics does not require expensive equipment to establish a controlled environment for data capture like iris biometrics [16]. This makes it simpler and cheaper to integrate into devices that are used on a daily basis. Moreover, commercial companies and engineers found fingerprint biometrics to be more accepted by the public than using iris biometrics [16, 22].

Of course none of these innovative authentication devices would have been possible if Galton did not discover the level of individuality of fingerprint biometrics in the 1890s [17, 23–25]. It was discovered that the probability of two different fingers having the same fingerprint was as small as $1 : 10^{-20}$ [26, 27]. Even though fingerprints were assumed to be unique to each individual at the time [23–25, 27], it had not been proven conclusively until this breakthrough. Fingerprint biometrics could only be used in criminal identification after this was statistically proven and it was discovered to be an optimal means of identification even when compared between twins. The criminal identification divisions previously used a cumbersome bertillonage technique which recorded the height of an individual as well as other physical measurements to verify identities of criminals [23, 27, 28]. It was only when the William West case occurred in Kansas [23, 28], that Scotland Yard began to rely on fingerprints for identification since it was the only human characteristic at the time that could differentiate between individuals who have similar physical features [23, 29]. Based on the amount of success brought about through fingerprints in terms of convicting criminals, the US congress decided to create a new unit within the Federal Bureau of Investigation (*FBI*) department, which now holds the central repository of all fingerprint records in the US [1].

1.2 Fingerprint classification

In 1924, the original number of fingerprints contained in the *FBI's* database (*DB*) was approximately 810 188 [1]. These were created using ink fingerprints on paper cards. Owing to the size of the *DB*, fingerprint records had to be filed systematically in order for employees to easily and quickly search through them on a daily basis. To reduce the number of manual searches, fingerprints which have the same global structure, known as a class, were grouped together [23, 25]. Incoming records were only compared to fingerprints of the same class. This search strategy is referred to as exclusive fingerprint classification, which stems from the Henry system of classification [1, 23]. The Henry system defines fingerprints into eight distinct classes [23, 25]; these consisting of a Whorl (*W*), Twin Loop Whorl (*TLW*), Plain Arch (*PA*), Tented Arch (*TA*), Left Loop (*LL*), Right Loop (*RL*), Central Pocket Whorl (*CPW*) and Accidental Whorl (*AW*) [23, 30].

The distribution of these fingerprint classes in nature is not uniform [18]. *CPWs*, *TLWs* and *AWs* are very sparse so they are frequently disregarded for classification purposes [18]. The likelihood of the other classes occurring are approximately 3.7%, 33.8%, 31.7%, 2.9% and 27.9% for a *PA*, *LL*, *RL*, *TA* and *W* respectively [18, 31]. Therefore, these five classes are commonly used today [19, 32–34]. The patterns of each of the five common classes are depicted in Figure 1.1.

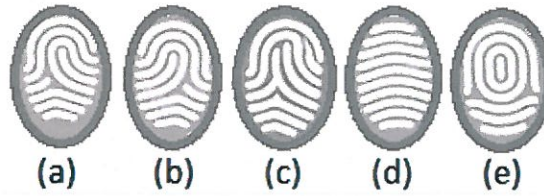


Figure 1.1: The five most common classes: (a) *RL*, (b) *LL*, (c) *TA*, (d) *PA*, and (e) *W*

1.3 Moving towards fingerprint automated systems

As a result of military documentation that was captured during World War II and the Korean conflict, the *FBI's* criminal and civilian fingerprint *DB* grew to approximately 78 million records [1]. Even though the number of comparisons was reduced using exclusive fingerprint classification, the process became labour-intensive and time consuming. Therefore, in 1963 special agent Cart Voelker of the *FBI* identification division and the National Institute of Standards and Technology (*NIST*) engineers attempted to solve this problem [1]. The contribution from their work and the projects developed by the head office in the United Kingdom, the Police department in Paris and the Japanese National Police, brought about the first Automated Fingerprint Identification System (*AFIS*) [1, 25]. Since the process was now automated and the *DB* contained scanned ink fingerprints on papers, the system had to automatically identify the classes to perform exclusive fingerprint classification. To do this, global features found on the fingerprint were used to detect a class type [18, 25]. The six commonly used features to achieve this are, Ridge-Structure features, Frequency-Domain features, Orientation features, Singular Points (*SPs*), FingerCode features and Structural features [18].

1.3.1 Brief description of fingerprint global features

A short definition of each of the six features are provided below:

Ridge-Structure

This feature involves using the ridge information directly or by representing all the ridges as a flow pattern to classify the fingerprint. Ridge structures are categorised into Fiducial Lines, Geometric Framework, Ridge Recurrence and Hierarchical Kernel Fitting [18, 35, 36].

Frequency-Domain

Fingerprints are composed of parallel and periodic lines which provide sufficient information to be applied to transforms for classification purposes [37–39].

Orientation

Fingerprints can be represented by the global orientation flow patterns [5, 38, 40–42]. The flow pattern is composed of individual orientation fields that represent a $w \times w$ block of pixels in the image [43]. Each class has a unique orientation flow [10, 44].

Singular Points

Exclusive classification techniques using this feature is commonly based on the number and type of each Singular Point (*SP*) found on the fingerprint [22, 33, 44–51]. There are two types of *SPs*, namely loops and deltas. Loops are the highest ridge curvature point, whereas deltas are the point where ridges triangulate, as illustrated in Figure 1.2. Fingerprint experts use these features to manually classify fingerprints, since this feature is one of the simplest and easiest to recognise. Furthermore, *SPs* forms the basis for many other features [22, 34, 37].

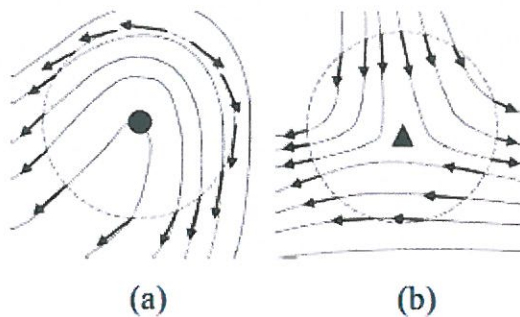


Figure 1.2: Singular Points (*SPs*) namely: (a) loop and (b) delta [44]

FingerCode

FingerCode is a unique feature vector that represents a single fingerprint image as four circular disks that result from using different Gabor filter parameters [22, 51]. These disks can be used to identify a class, since each class produces unique disks.

Structural

The orientation fields are partitioned into homogeneous regions [4, 52, 53]. An example of a structural feature, is directional patterns (*DPs*) which are partitioned into orientation fields falling within the same range. Each orientation partition is referred to as a region. The arrangement of these regions represent the structural feature of a particular fingerprint. These arrangements differ for each class and classification can be easily performed. One of the common ways to perform the classification is by using Graph-base techniques.

1.4 Emergence of live fingerprint scanners

Studies conducted during 1986 to 1988 showed that by using the *AFIS* the conviction rate into state prisons for burglary cases was three times higher than burglary cases which did not use *AFIS* [1], as depicted in Figure 1.3. It was only when the *AFIS* became successful in producing high conviction rates into state prisons, that a great deal of attention was directed towards the development and improvement of the fingerprint technology [1]. Owing to it's success during this era, fingerprint technology was rapidly advancing.

By 1999, the *AFIS* was widely used by 500 *AFIS* sites worldwide [1] and was further exposed to commercial companies through exhibits at International Association for Identification *IAI's* conferences [1]. It was through these conferences that the government become aware of the digital technology which was being produced by commercial companies [1]. The government began investing towards new developments related to the combination of digital image enhancement and forensic science [1]. Therefore, not long after the first digital imaging system was introduced, the San Francisco Police Identification Bureau was equipped with the first live scan fingerprint device [1].

Commercial companies found great potential in these devices and were willing to contribute to further development [18]. Shortly after, fingerprint live scan devices were commonly integrated

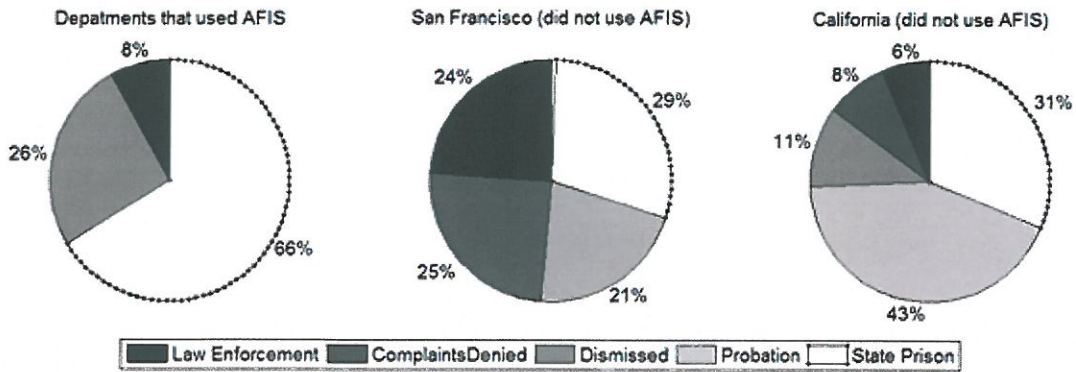


Figure 1.3: The success of tracking fingerprints using *AFIS* during 1986 to 1988 [1]

into door alarms, laptops, computer keyboards, and even USB devices. However, for these uses, devices had to be made more compact, meaning that less area was captured compared to the original ink-rolled fingerprint on paper method [54, 55].

1.5 Motivation

Owing to the size of flat fingerprint live scanners, fingers are easily incorrectly placed on these devices resulting in incomplete fingerprints which often exclude key features such as *SPs*. A fingerprint with a certain number of loops and deltas are referred to in this dissertation as a case. Figure 1.4 shows an example of flat fingerprint cases containing only a loop for class (a) *RL*, (b) *LL*, and (c) *TA* where the entire region below the loop is not captured. *RL*, *LL* and *TA* look highly similar. Therefore, these incomplete fingerprints that are missing a delta generates small inter-class variability that leads to misclassification [10, 19, 44].



Figure 1.4: Small inter-class variability issues between class (a) *RL* (b) *LL* and (c) *TA*

The small inter-class variability challenges are worsened by rotation. When rotation of a fingerprint occurs even as little as approximately 10° , a partial/incomplete *LL* such as Figure 1.4 (b), can then resemble a *RL* shown in (a) or *TA* in (c). This may result in the entire *DB* being searched to find a possible fingerprint match. *DBs* are considerably large often containing millions of fingerprint images. On that account, when the entire *DB* is searched, the response time of the fingerprint recognition system increases. Many practitioners handling these cases generally discard these fingerprints from the testing sets [19, 56]. However missing *SPs* is a common challenge that results from flat fingerprint scanner devices and discarding the images when testing algorithms does not benefit the system. Discarding the images in real *DBs* cannot be easily accomplished and since *DBs* contain millions of records which are captured over decades, it makes it difficult to recapture the fingerprints regardless of the type of scanner used. Even by using a different scanner that is less likely to capture incomplete fingerprints with missing *SPs* (e.g., an electronic rolled fingerprint scanner), does not guarantee that the end user who is enrolling the fingerprint correctly places the finger on the device. Therefore, solving this from the software side is more logical since using bigger scanners may not necessarily solve the problem. To overcome these factors and prevent misclassification, a classification technique that is capable of handling the challenges of missing *SPs* can be created instead. Previous fingerprint classification techniques that have proposed to solve the challenges of missing *SPs*, have used local orientation and the remaining *SPs* [10, 19, 44]. However, not all cases were accounted for and some rules did not preserve rotation [10, 19, 44].

1.6 Problem statement

The problem with compact scanners is that the fingerprint is not fully captured and this results in missing *SPs* [10, 55]. These flat fingerprints are difficult to classify due to the amount of information captured [10]. Researchers often use local orientation features since the maximum amount of information of the fingerprint can be extracted [10, 19, 44]. Many classification methods track the local orientation fields around the remaining *SPs* [10, 19, 44, 55]. Despite their success of achieving a good classification accuracy when classifying fingerprints with missing *SPs*, methods that track local orientation fields experience difficulties in differentiating between

classes that have hardly any information available especially under noise. With these problems not all flat fingerprint cases can be solved using one research technique [10, 19].

Previous classification techniques have shown that each fingerprint feature has different benefits when classifying complete fingerprints [7]. Since flat fingerprints are a recent research topic and classification techniques thus far have only used local orientation features to classify these cases [10, 19, 44], finding an optimal feature has not been covered. Therefore, to classify the more difficult cases and prevent further small inter-class variability issues, the capabilities of each feature have to be re-evaluated using previous literature. The literature study will evaluate the advantages and disadvantages of each fingerprint feature used in exclusive classification to establish the optimal features that can be used to solve the challenges of missing *SPs* in fingerprint classification. This research will investigate whether using the selected feature can successfully classify flat fingerprint cases. The investigation can be assessed by developing a flat fingerprint classification algorithm that is resistant to missing *SPs*.

1.7 Research question

This research attempts to produce new knowledge to answer the following question:

1. Can flat fingerprints with missing *SPs* be successfully classified by using the most appropriate fingerprint features?

1.8 Research goal

The aim of the research is to classify flat fingerprints with missing *SPs* by using the most appropriate fingerprint features.

1.9 Research objectives

The objectives of the research are as follows:

1. In the literature study, determine which are the optimal features that can classify flat fingerprints with missing *SPs*.
 - Establish the advantages and disadvantages of each fingerprint feature (Ridge-Structure feature, Frequency-based feature, Structural feature, *SPs* and Orientation features).

2. Achieve unique and consistent representations of the selected feature for each flat fingerprint case.
 - Evaluate the changes in the selected fingerprint feature under rotation and other influential factors.
3. Design an exclusive fingerprint classification that is based on a rule-based technique which classifies flat fingerprints with *SPs* using the selected features.
 - Develop classification rules based on the unique and consistent representations of the selected feature for each flat fingerprint case.

1.10 Delineation

The bounds of this research are as follows:

1. The Fingerprint Verification Competition (*FVC*) *DB* is used to test the fingerprint classification algorithm. This *DB* contains images captured from flat fingerprint live scanner devices. Since the main focus of the research is to solve flat fingerprint challenges with issues of missing *SPs*, this is the most suitable *DB*.
2. Only exclusive fingerprint classification using a rule-based approach is considered.
3. Existing pre-processing methods will be used where required.

1.11 Contributions

This research makes four main contributions to the field of biometrics in exclusive fingerprint classification:

1. It presents a summary of the advantages and disadvantages of each fingerprint feature (Ridge-Structure feature, Frequency-based feature, Structural feature, *SPs* and Orientation features) when used in exclusive fingerprint classification.
2. Directional Patterns (*DPs*) is one of the selected features that is chosen based on the literature study. Therefore, an evaluation of the *DP* of each flat fingerprint case under the variation of the number of regions and rotation is conducted. This establishes the ideal number of regions to form a *DP* and method of rotation used to produce unique and consistent *DPs* for each flat fingerprint case.

3. The research presents a novel rule set for exclusive fingerprint classification using *DPs* and the remaining *SPs* that classifies flat fingerprint cases.
4. An attempt is made to solve the challenge of classifying a *PA* and a Partial Fingerprint (*PF*) by creating a more robust rotational invariant rule that does not rely on the fingerprint's size but rather on the actual class pattern.

1.12 Brief chapter overview

Chapter 2 covers the literature review that identifies the most appropriate feature to classify fingerprints with missing Singular Points (*SP*). Chapter 3 presents the unique and consistent representations of the selected feature (i.e., *DPs*) for each flat fingerprint case. The rule-sets are developed on the unique and consistent representations of each flat fingerprint cases. The classification rule set is covered in Chapter 4. Results of the proposed classification algorithm will be presented in Chapter 5. Chapter 6 provides the conclusion.

Chapter 2

Literature Review

This chapter discusses the works in literature which cover the fingerprint features used to implement exclusive fingerprint classification algorithms.

2.1 Introduction

During the last century, practitioners have largely contributed towards the development of inexpensive and compact flat fingerprint scanners to meet public demand [55, 57, 58]. These hardware constraints limit the amount of information captured from the finger [57, 58]. These fingerprints are generally placed flat on the scanner and are not rolled. The *SPs* found on the fingerprints are therefore frequently excluded [19, 44]. In this dissertation a case is referred to a fingerprint of a certain number of loops and deltas. Often, there are cases where classes look very similar because they have the same number of *SPs* and the ridge information is not enough to differentiate between the classes [10, 44].

Initial fingerprint classification techniques were not capable of handling cases of missing *SPs* [4, 7, 9, 10, 22, 33, 53, 56, 59], since they were designed for complete fingerprints that experience at most minor ridge loss [22, 37, 53]. The inter-class variability issues across classes, caused by missing *SPs* were difficult to overcome and as a result, misclassification often occurred [10]. This can lead to a match not being found in the sub-set. The consequences of not finding a fingerprint match within a sub-set of the *DB* that holds the detected class types leads to the entire *DB* being re-searched for a possible match. This increases the response time of the recognition system. To prevent any occurrence of misclassification, practitioners refocused their attention on the development of classification algorithms specifically designed to cater for the downfalls of flat fingerprints [7, 10, 19, 44, 55].

In 1999, the *FVC* released the first large *DB* consisting of digitally scanned fingerprints [57]. The testing data that was made available was sufficient to identify the key problems of flat fingerprints [57]. Researchers were then able to begin developing classification algorithms for current technology [57]. Not only were there concerns regarding lack of information captured from optical scanners, but rotation, translation and noise were additional factors that contributed heavily to interclass variability across fingerprint classes [10, 44].

Each feature has different capabilities and selecting the most appropriate features can help overcome the challenges faced by flat fingerprints [7, 44, 55, 60]. Therefore, to develop the fingerprint classification algorithm for this dissertation, the feature will be selected based on its ability to handle minimal information under noise, rotation and translation. A full literature study of previous classification techniques using different features, namely: *SPs*, FingerCode, Structural feature, Ridge-Structure feature, Frequency-based feature and Orientation features was carried out based on each feature's ability to handle noise, computation, rotation, translation, and inter-class variability issues, but mainly missing *SPs*. The study analysed all classification techniques regardless of the type of approach from all periods in order to produce a sound analysis of the capabilities of the features.

The ideal way to analyse the features would be to implement all classification techniques and test them on images with minimal information. However this requires a larger scope encompassing different enhancement and machine learning techniques covered over the past century. Therefore, when determining the optimal features that can be used to solve missing *SP* challenges, obtaining a generalised critical analysis of the past literature is sufficient.

A detailed literature study of classification techniques including both rule-based and machine learning approaches of the selected features will be covered. This will allow the most optimal feature to be selected and underline the benefits and downfalls of the feature chosen, in order to determine the most appropriate additional features to improve the techniques classification accuracy.

Classification techniques that have attempted to classify fingerprints with missing *SPs* will also be discussed to establish the current gaps. This literature will cover the period from 1999 to 2015, since the *FVC DB* was only released in 1999 and therefore research on flat fingerprints gained momentum during this period [7, 10, 19, 55, 57]. The literature review highlights the shortcomings of these works and then highlights possible solution directions.

The following sections present the literature study that identifies the most appropriate feature

to classify fingerprints with missing *SPs* caused by compact electronic scanners. Section 2.2 introduces the summary of the advantages and disadvantages of all features used for fingerprint classification and highlights which features show potential for solving cases of missing information. The details of the selected features are also presented in this section. Section 2.3 provides literature covering fingerprint classification methods during the period from 1999 to 2015, that attempt to solve classification challenges of flat fingerprint cases. The major gaps will be addressed in Section 2.4 and based on previous literature a possible solution is presented. Section 2.5 discusses a fingerprint classification technique introduced in 2008, that first use this suggested features and highlights the gaps to be addressed by this work. Conclusions are drawn in Section 2.7.

2.2 Selecting the optimal features to overcome missing *SPs*

In this section, the possible optimal features that could be used to reduce the challenges faced by flat fingerprints are determined by examining existing literature.

2.2.1 Advantages and disadvantages of fingerprint features

Previous fingerprint classification techniques have used features such as: *SPs*, FingerCode, Structural features, Ridge-Structure features, Frequency-based features and Orientation features. These techniques are analysed to establish the advantages and disadvantages of each fingerprint feature. Table 2.1 shows a summary of the advantages and disadvantages of the various features. The ability to handle missing information, rotation, translation, inter-class variability issues and computational cost are observed based on the accuracy of the algorithm; the way in which the features are used; types of images excluded from the testing process; requirements and types of enhancement techniques used; dependency on rotation; and consistency of patterns.

Table 2.1: Summary of the Advantages and Disadvantages of Ridge-Structure features, Frequency-based features, Orientation features, SPs , FingerCode and Structural features

Feature	Advantages	Disadvantages
Ridge Structure	<ul style="list-style-type: none"> -Invariant to translation [61] -Invariant to rotation [61] -Contains all information about a class [5, 61] 	<ul style="list-style-type: none"> -Sensitive to noise [5, 61] -Requires enhancement techniques [61] -High computational cost [5, 61]
Frequency-based	<ul style="list-style-type: none"> -Invariant to noise [37, 62] -Invariant to rotation [37, 62, 63] -Invariant to translation [37, 62, 63] 	<ul style="list-style-type: none"> -Small inter-class variability between PA and TA [37, 62] -Small inter-class variability between RL, LL and TA [37, 62] -High computational cost [37]
Orientation	<ul style="list-style-type: none"> -Handles poor quality images [18], however it fails when there is minimum ridge information [32] -Handles missing information such as excluded SPs [44] -High inter-class variability between LLs, RLs, TAs and Ws [9, 32, 44, 64] -Invariant to translation [44] -Invariant to rotation, however dependent on how its used [44] 	<ul style="list-style-type: none"> -High dimensionality [9, 18] -High computational cost [9, 18, 64] -Small inter-class variability between PAs and TAs [9, 44]
SPs	<ul style="list-style-type: none"> -Invariant to translation [33, 65] -Invariant to rotation [33, 65] -Lower computational cost compared to Orientation and FingerCode features [59] -Simple efficient method to classification [19, 56] 	<ul style="list-style-type: none"> -Small inter-class variability between TAs, RLs and LLs [5, 19, 34, 56] -Do not work best in poor quality images [33, 44, 50, 56, 66] -Low ability to handle missing SPs and requires features like orientation to improve the method [19, 44]
FingerCode	<ul style="list-style-type: none"> -Highly robust to noise [22, 48, 51] -Invariant to rotation [22, 48, 51] -Invariant to translation [22, 48, 51] 	<ul style="list-style-type: none"> -High computational cost [22, 48, 51] -Difficult to determine registration point [22, 51] -Small inter-class variability between Ws, RLs and LLs [22, 51] -Accuracy dependent on the classifier [22, 51] -Difficult to train Ws due to inherent ambiguities [51] -Low ability to handle missing SPs [22]
Structural	<ul style="list-style-type: none"> -Able to handle ridge loss [4, 5] -Invariant to rotation [4, 5] -Invariant to translation [4, 5] 	<ul style="list-style-type: none"> -Uses classifiers for classification [3, 4, 30, 52, 53, 67] -Small inter-class variability between PAs/TAs and RLs/LLs [4, 5, 53] -Accuracy depends on how well the orientations are grouped into homogeneous regions [4, 53] -Affected by noise [5, 30, 53]

2.2.2 Orientation feature

Based on the information formulated in Table 2.1, orientation features have many advantages and are one of the features that best represent a class [7, 25, 68], since maximum information about the fingerprint class can be extracted without the excess computational cost required for other features, such as ridge-structure features [30, 55]. Even as early as the 1970s, Kaas *et al.* stated that orientation field patterns have the ability to describe the most complex images, regardless of shape and colour, as long as the magnitude and direction is known [69]. Owing to these factors, orientation features are generally the most appealing feature to develop a successful classification technique [7, 68].

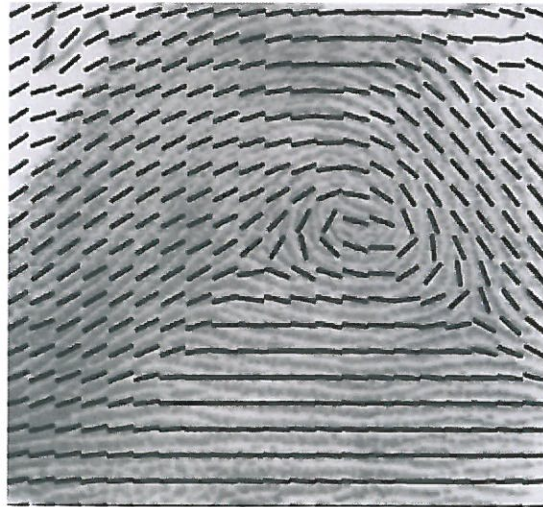


Figure 2.1: W fingerprint represented by block-wise orientation fields [2]

Additionally, orientation features are not as sensitive to noise as ridge-structure features [30, 55, 68]. This is due to the fact that orientation is often averaged over a block of ridges [55], as shown in Figure 2.1. The blocks of orientation field reduce the effects of corrupted images [55, 70]. This invariancy to noise makes orientation more appealing to classification of flat fingerprints [30, 55, 68]. Furthermore, the orientation fields are small enough to track ridges when minimum information is captured [10, 44, 68]. The actual fields are also more reliable than ridge-structure features [30, 60].

The reliability and potential capabilities of orientation fields, is clearly demonstrated in one of the most popular classification techniques to date, known as the Pattern-level Classification

Automation System (*PCASYS*) method which was developed in 1995 [3–5, 34, 35, 71]. It was the first open source fingerprint classification method [9, 18, 72].

However, a downfall of using of orientation feature vectors is that they can lead to unnecessary dimensionality and computational cost [18]. The Karhunen-Loeve transform was applied to the *PCASYS* model to reduce the high dimensionality from 1680 elements to 64 elements [9, 18]. However, even though dimensionality was improved further by using alternate machine learning techniques namely: Self-Organizing Map (*SOM*) techniques [18, 73, 74], two stage classification techniques [71], or Support Vector Machine techniques [70], the effect of the dimensionality reduction increased the computation cost [18]. These factors made it difficult to integrate the method into fast performing scanner devices [30].

2.2.3 Benefits of combining SPs with orientation features

Unlike using orientation fields alone, combining *SPs* with orientation fields can reduce dimensionality and computation cost [46, 47, 56, 75]. Rather than using the orientation fields of the entire image, only the flow of orientation fields between the loop and delta region was used to classify complete *RLs*, *LLs* and *TAs* [10, 19, 46, 56, 75].

The work of Karu and Jain produced one of the simplest solutions using orientation fields and *SPs* [56]. The classification of *RLs*, *LLs*, *TAs*, and *Ws* is based on the threshold orientation value which is computed using the fields along the line that is constructed between the loop and the delta [56]. Speed tests showed that this work was quicker than *PCASYS*, taking approximately 3 seconds to perform overall classification on a Sparc-20 workstation [56] as shown in Table 2.2. The computation of the orientation fields took most of the time (2.8 seconds), and the actual classification was then completed in just 0.2 seconds [56]. Conversely, the orientation feature vector based *PCASYS* algorithm had a minimum running time of 5 seconds on the HP and 16.7 seconds on the Sun SPARC-station (10), as shown in Table 2.2 [9]. This illustrates that *SPs* have considerably lower computational cost with no problems related to dimensionality, making it much easier for integration with modern scanner devices. The *PCASYS* was tested on 2699 images from the *NIST* Special *DB* 14 and the method by Karu and Jain was tested on the entire *NIST* 4 *DB* which consisted of 4000 images [9, 56].

Additionally, the key information about a class lies around the *SPs* [10, 33, 44]. Of course, *SPs* cannot be used alone in classification [10, 19, 44, 56], since the amount of information would not be enough to over interclass variability issues, especially for flat fingerprints [10, 44, 46].

Table 2.2: Speed results for *PCASYS* model and a fingerprint classification model based on *SPs* [9]

Author	Computer Model	Seconds
<i>PCASYS</i> [9]	Dec Alpha	7
	Hp 9000/735	5
	IBM 7012/370	6.7
	SGI Challenge (IP19)-One processor	7.5
	Sun SPARC station 2w/Weitek 80 MHz CPU	17.7
	Sun SPARC station 10	16.7
	Sun SPARC server 4/470	32.9
Method by Karu and Jain [56]	SPARC station 20	3

Nonetheless, based on the advantages of using the remaining *SPs*, they are the most suited feature to be integrated with other features to overcome the challenges of flat fingerprints.

The combination of these features also aids with issues related to translation and rotation [10, 44, 56]. Since orientation fields are calculated using a set coordinate axis [43], large variation of fingerprint patterns occur under rotation and translation. *SPs* [33, 76] are easily recognisable fingerprint landmarks and therefore are commonly selected as registry points [33, 76]. Methods that use orientation fields directly for classification [70], like *PCASYS* [9] and *SOMs* [77]; or indirectly like frequency [22], ridge-structure [5] and structural based models [4] have all used *SPs* as an alignment tool or to locate specific areas on a fingerprint [18]. Since they are such a good landmarks, even for fingerprints with missing *SPs* the remaining *SPs* can also be used to align the fingerprints to obtain unique and consistent orientation patterns for each class [7]. The remaining *SPs* found on the flat fingerprint cases can also be used as registry points to easily locate the regions of interest that will provide the most information about the class type [10, 44, 56].

These factors serve as a justification on how the combination of orientation feature and *SPs* compliment each other's strengths and weakness, resulting in them being the preferred features to overcome the challenges of flat fingerprints.

2.3 Fingerprint classification method designed to overcome the challenges of missing SPs

When *FVC* released the first publicly available *DB* of electronically captured fingerprints in 1999 [57], it allowed researchers to bridge the gap between industry and academia [57]. Practitioners could make improvements on fingerprint classification techniques specifically designed to overcome the challenges of flat fingerprints, which is the type of fingerprint captured in most modern applications [10, 44, 55, 57].

Tests conducted on the *FVC DB* allowed practitioners to easily identify which aspects needed attention [10, 44, 55, 57]. Most older methods that were tested on the *NIST DB* were not designed to cater for electronically captured fingerprints [56]. An example was the method by Karu and Jain which was originally tested on the *NIST-4 DB* which contained 4000 scanned images of ink fingerprints that were rolled on paper. It produced a classification accuracy of 85.4% [56]. In spite of that, when this work was re-implemented and tested on the *FVC 2002 DB* and *FVC 2004 DB*, the accuracy dropped to 47.3% and 55.0%, respectively [10]. The technique used simple rules that combined the orientation feature with *SPs* [56], however the method was too dependent on both the loop and the delta to determine a class. The *FVC DBs* contain fingerprint images with missing *SPs* and therefore the drop in accuracy depicted that the original rule-based strategy was not capable of handling flat fingerprints with missing *SPs* obtained from electronically fingerprint scanners [10]. Researchers therefore began to develop techniques to accommodate all cases of missing *SPs*, otherwise fingerprint classification algorithms fall short when flat fingerprints are not completely captured [10].

Table 2.3 covers all possible cases of captured *SPs*, including missing *SPs*, that can occur and the possible class types. It also covers all techniques designed to classify each case. The table illustrates which cases have been addressed in the literature and which have not been resolved yet.

Since a fingerprint image with a single loop captured [10, 19] and a *W* with only two loops or two deltas captured [19, 78, 79] were the most frequently occurring cases of missing *SPs*, many techniques have attempted to address these [10, 19, 78, 79]. The most simplest of all methods, was a rule-based classification technique developed by Msiza *et al.* [19]. This technique used coordinate geometry of *SPs* [19]. The rules caters for classifying a single loop image that can

Table 2.3: All possible flat fingerprint cases including missing *SP*s and the corresponding literature that attempts to solve each case [10]

No. Loop	No. Delta	Fingerprint Class	Examples in literature that account for each case
2	2	<i>W</i>	[10, 19, 44, 56, 78, 79]
2	1	<i>W</i>	[10, 19, 44, 78, 79]
2	0	<i>W</i>	[10, 19, 44, 78, 79]
1	2	<i>W</i>	[10, 19, 44, 56, 78, 79]
0	2	<i>W</i>	[10, 19, 44, 56, 78, 79]
1	1	<i>W</i>	x
		<i>TA</i>	[19, 56, 78, 79]
		<i>RL</i>	[19, 44, 56, 78, 79]
		<i>LL</i>	[19, 44, 56, 78, 79]
1	0	<i>W</i>	[10, 44]
		<i>TA</i>	x
		<i>RL</i>	[10, 19, 44]
		<i>LL</i>	[10, 19, 44]
0	1	<i>W</i>	x
		<i>TA</i>	x
		<i>RL</i>	[44]
		<i>LL</i>	[44]
0	0	<i>PA</i>	[10, 19, 56, 78, 79]
		<i>PF</i>	[10]

be considered as a *LL* or *RL*. The method uses the location and the angle between the loop and the pedestrian plane to identify a *LL* from a *RL*. A rule for the detection of a *W* with missing delta points was also created.

The additional rules by Msiza *et al.* that were created to cater for missing *SP* cases, produced a higher classification accuracy than the one achieved by the work of Karu and Jain, which only addressed complete fingerprint cases [10, 56]. The accuracy increased by 19.6 % and 18.0% when re-implemented and applied on same batch of images from the *FVC 2002 DB1* and *FVC 2004 DB1*, respectively [10, 19]. However, the classification accuracy was still low [10] which drew attention on the fact that improvements with regards to classifying fingerprints with missing *SP*s still had to be made. The misclassification of fingerprints were owing to the method's inability to handle rotated fingerprints and rules were not extended to address all cases of missing *SP* [10].

In 2014, the works of Webb *et al.* [10] and Guo *et al.* [44], used local orientation features to produce a more robust set of rules for classification of a single loop fingerprint. The rules were invariant to rotation, since they were not dependent on the location of the *SP* or an angle that was

calculated using a fixed x-y axis [10, 44]. The orientation fields were observed to differentiate not only *LL* and *RL*, but also for *W* classes [10, 44]. Although the rules for each work were fairly different, both were based on the information provided by flow of orientation fields around the loop, which moved towards the left, right and converged for a *LL*, *RL* and *W*, respectively [10, 44].

Webb and Mathekgga even introduced a new rule that detected a Partial Fingerprint (*PF*) from a *PA* [10]. *PF*s are the most difficult fingerprints to classify, since they have no information that can be used to determine the class type. These fingerprints are cut-off before the class pattern resulting *RL*s, *LL*s, *TAs* or even *Ws* being captured with no *SP*s. Considering that both *PAs* and *PF*s have no *SP*s, *PF*s are commonly classified as *PAs*. The work of Webb and Mathekgga observed the size of the fingerprint to identify a *PF* from a *PA* [10]. The downfall with using size of the fingerprint captured, is that the size of complete fingerprints vary for different individuals. Guo *et al.* also presented a new rule for single delta fingerprints. This was established by tracking the deltas locations relative to the centre of the print, to classify a *LL* or a *RL*. However, this rule did not preserve rotation [19]. In summary, these rules attempt to solve the challenges faced in fingerprint classification, which have not been addressed until now, but there is room for improvement.

2.4 Limitation of local orientation fields

Guo *et al.* mentioned that the major contribution for classification accuracy not reaching 100%, was the effect of noise on local orientation features [44]. To overcome this downfall, local orientation classification methods including the work by Guo *et al.* , perform fingerprint enhancement [2, 44, 60, 78]. This helps to reduce excess amounts of noise that results in the restructuring of ridges [58].

Nonetheless, errors still occurred in both complete and partial *PA* and *TA* classes, due to the sensitivity of local orientation flow. The *PA* and *TA* fingerprints were often classified as a *RL* or *LL*. The averaging of orientation fields reduces the risk of the impact of noise to an extent, however small inter-class variability issues between classes with a similar number of *SP*s or patterns cannot be avoided under the influence of noise [44].

2.5 Representation of classes using global orientation fields

Observing smaller decomposed segments of an image, can make it difficult to differentiate corrupted data from true data. Rather than focusing on enhancement techniques to prevent corrupted data, globally analysing any image can provide a perspective of its layout that is not affected by local noise. Thereafter, small areas of noise become less of a concern [53, 55].

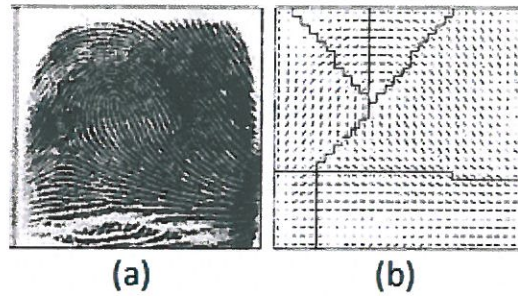


Figure 2.2: An illustration of a (a) *RL* and it's (b) global pattern (structural representation) with four homogeneous regions [3]

The fingerprint must be represented as a structural identity to observe the orientation fields globally. The Structural Approach introduced in 1996, demonstrated its ability to overcome misclassification due to the effect of corrupted fingerprints [53]. Therefore, this may be a solution to further increase classification accuracy on flat fingerprint cases. The approach grouped orientation fields into homogeneous regions to form a global pattern for each class. An example of a *RL* global pattern with four homogeneous regions is shown in Figure 2.2 (b).

To overcome misclassification using structural features, the type of region segmentation method that groups the orientation fields plays a significant role [4]. It affects the patterns of classes [4, 52, 53, 80] or consistency of the number of regions produced [4, 52, 53, 80]. For images with missing *SPs*, patterns have to be consistent and unique for each class, so that the inter-class variability issues can be dealt with [7]. Therefore, determining the region segmentation method is crucial.

Maio and Maltoni's structural approach was designed specifically to minimize the variance of the orientation fields, so that it can establish regularity of the homogeneous regions [53]. The regions are segmented using a greedy dynamic clustering algorithm [53]. However, the problem with the greedy dynamic clustering algorithm is that it produces inconsistent patterns [4].

Moreover, the technique's final stage is based on graph matching to detect the closet resemblance of the graph model to the model associated to a particular class [53]. However, using the model node graphs requires complete class information which is not possible for flat fingerprints [53]. Images from live-scan devices have often shifted, cutting off large portions of regions which will result in minimum information and translational issues [55]. In addition, Cappelli *et al.* claimed that the method was too dependent on ridge-line orientation changes on achieving homogeneous regions and the dynamic concept of clustering had inconsistency of the number of regions [4]. It gave rise to the hidden similarities across classes and the huge differences between the same class. Figure 2.3 shows a *LL* that produces different number of regions using the greedy method [4]. Such a case was seen in the work of Yao *et al.* , where a total of twelve models had to be created for just a five class classification [52].

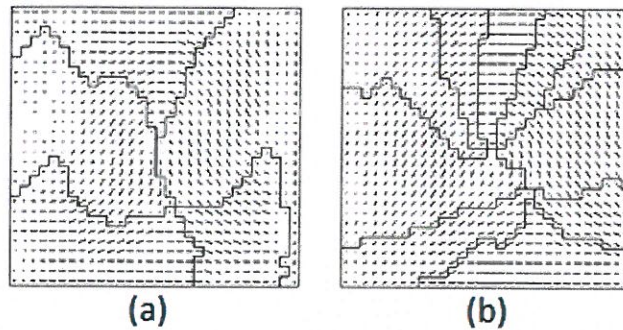


Figure 2.3: Structural representation of *LL* fingerprints using the greedy method which produces (a) three regions and (b) six regions [4]

One way to decrease the degree of freedom of the number of regions is to apply a guided segmentation, where the number of regions and their structural representation are fixed [4]. Each class is represented by a dynamic mask [4]. The problem with this technique, is that incomplete fingerprints do not have enough information to apply a guided segmentation on the orientation map. Hence, overall graph matching approaches using either homogeneous regions or dynamic masks would not be able to perform successfully on flat fingerprints.

Another possible region segmentation method which requires less computational cost than the presented techniques, creates regions by combining the orientation fields that lie within a specific orientation range [5, 6, 80–82] to form what is known as a directional pattern (*DP*), as shown in Figure 2.4. The key aspects that set this technique apart from others, is that the variation of

patterns for an upright fingerprint is limited, since the number of region is consistent [5, 6, 80–82]. In addition, to perform classification it does not depend on complete fingerprints to associate it to a graph model [5]. The information from the pattern is sufficient to classify a fingerprint directly using a rule-based approach [5, 80].



Figure 2.4: Example of a (a) *TA* and its (b) *DP* [5]

2.6 Directional pattern approach

In 2008, the first rule-based classification technique using *DPs* was introduced [5]. The directional approach is implemented using sets of pattern descriptors which are divided into 18 pattern zones to classify fingerprints [5], as shown in Figure 2.5. Figure 2.5 (c) shows the associated single loop and single delta descriptor, each with two zones that represent a specific pixel direction [5]. The descriptor is designed for *DPs* that are formed from fingerprints in which the *SPs* have been horizontally aligned [5]. The use of pattern descriptors, implies that the *DP* for each class produces consistent unique patterns irrespective of the angle of rotation of the fingerprint captured [5].

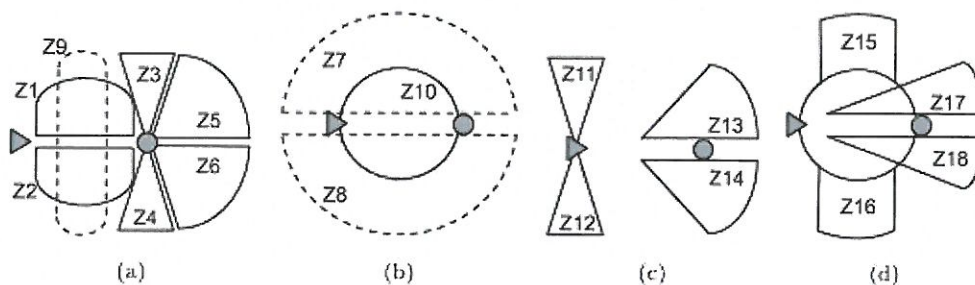


Figure 2.5: Eighteen pattern zones with deltas and loops [5]

Even though the approach was designed for fully rolled fingerprint, Lui *et al.* extended the rules to cater for the most frequent case of missing *SPs* (a fingerprint with a single loop) [5]. The value of the pixel direction under the loop of a fingerprint with single loop is used to classify a *LL* and a *RL*. This highlights that *DPs* can globally represent the key information of a class regardless of the amount of information captured, if rotated correctly [5]. This is beneficial for fingerprints with missing *SPs*.

Conversely, if the pattern descriptors are extended for all missing *SP* cases, it will become complex and the approach will have an exceedingly large number of descriptors [7]. To prevent this from occurring while still utilising the concept of *DPs* for classification of most cases of missing *SPs*, the technique should not use pattern descriptors [7].

For this study a more simplified rule-based approach using *DPs* that are combined with *SPs* will be implemented to enable a successful flat fingerprint classification. This will use the advantages of orientation features and *SPs* for classifying flat fingerprints, but reduce the issue of noise experienced by local orientation fields. A rule-based approach has the advantage of not requiring a training data set. To produce consistent unique global patterns for each class, the *DPs* are observed under variation of rotation and number of region. The new rules will be created, based on the patterns obtained.

2.7 Conclusion

Compact electronic scanner devices that are widely used produce flat fingerprint images that are often not fully captured [55]. This leads to frequent exclusion of *SP* [10, 19, 44]. *SPs* combined with orientation fields did show potential but using local orientation fields suffered from noise. To counteract the lack of information and prevent small inter-class variability issues, *SPs* combined with *DPs* were selected to address the aim of the dissertation.

A previous directional approach which used pattern descriptors to successfully perform a rotational invariant classification drew attention to the benefits of combining *DPs* and *SPs* [5]. *DPs* were found to best represent a class, since the key information of a class is easily recognisable [5, 7]. The *SPs* also played an important role. The number and type of the *SPs* give an indication of what type of information is available and where on the fingerprint the 'region of interest' is located [5, 7].

The previous directional approach technique was not specifically aimed at addressing the challenges of flat fingerprints [5, 7]. Nevertheless, it did attempt to classify some cases with minimum information [5, 7], and therefore emphasised the potential it may have in solving flat fingerprint challenges. The purpose of this dissertation is to develop a fingerprint classification algorithm that is capable of classifying flat fingerprints with missing *SPs*. The following chapters will first investigate how to create the *DPs* such that it produces unique patterns for each flat fingerprint case, and then discuss the development of the classification rules.

Chapter 3

Evaluating the Change of Fingerprint Directional Patterns Under Rotation and Number of Regions

In this chapter, the evaluation of the changes in DPs are observed by varying the rotation of different fingerprints and changing the number of regions, to establish a way to consistently produce unique DPs for each flat fingerprint case.

3.1 Introduction

Fingerprint classes suffer from a great deal of small inter-class variability. This is worsened when there is missing information. To deal with the inter-class variability issues regardless of the type of features used, consistent and unique patterns that best represent each class type must be established. This work aims to use *DPs* for classification. Thus, the condition must be met for the *DPs* (i.e., unique and consistent). In this chapter the ideal conditions of *DPs* are evaluated to determine which conditions produce unique patterns. In order to do this, the factors that influence the *DP* class patterns first have to be established.

The most obvious factor that affects the patterns produced for each class is the number of regions that are used to form a *DP*. This is a controlled condition that is set by the developer. An example of a four region *DP* is shown in Figure 3.1 (a) and a nine region *DP* is shown in (b). These *DPs* are obtained from the same individual's fingerprint.

As shown in Figure 3.1, the size of the regions differ for different number of partitions. In addition, the changes in the number of regions affect the amount of visible noise present. As a

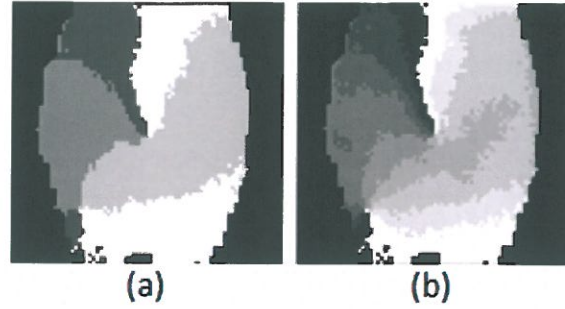


Figure 3.1: Example of (a) a four region *RL DP* and, (b) a nine region *RL DP* [6]

result of these observations, *DPs* with different number of regions should be analysed to establish which will generate the least amount of visible noise. Furthermore, the findings gathered from [6], stated that the intersection of fault lines (boundary lines of each region) of these *DPs* influenced the detection accuracy of *SPs*. Regarding Figure 3.1, when there are four regions, as in (a), the fault lines converge at one point, indicating a *SP*; however for nine regions it appears to converge at multiple points. Since the proposed classification will rely on the use of *SPs*, it is important to determine which number of regions produces the highest *SP* detection accuracy. Generally *DPs* are segmented into three [5, 7] and four [3, 53, 82] regions. Wang *et al.* claimed that a four region partition provides adequate information to locate points of intersection of all regions and the size of each region is large enough to easily determine a true *SP* [7, 82]. The authors also stated that *DPs* that have more than eight region partitions are more vulnerable to noise [82]. However, no supporting experiments were undertaken to validate these statements.

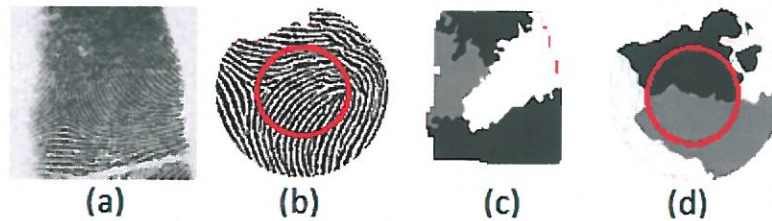


Figure 3.2: Examples showing, (a) an upright *RL*, (b) a rotated *RL*, (c) *DP* of an upright *RL* and (d) *DP* of a rotated *RL* [5]

Another factor that alters the patterns, is rotation. Considering that *DPs* are formed by grouping orientation fields that are calculated using a fixed x-y coordinate axis, the ridge directions of an individual fingerprint change as it is rotated. Hence, different patterns can be produced under

rotation, as illustrated using the work of Liu *et al.* [5]. Figures 3.2 (c) and (d) show two different *DPs* produced from the same class, that have been rotated at different angles.

On that account, a constant method of rotating the fingerprint needs to be found that will produce consistent and unique patterns of each class. Even though the classification techniques by Liu *et al.* [5] and Dorasamy *et al.* [7] used a specific method of rotation to obtain *DPs*, no supporting experiments were undertaken to justify why a specific method of rotation for fingerprints was selected.

Overall, the literature that used these specific types of *DPs* have not conducted any explicit research to investigate the best possible conditions of *DPs*, both in terms of rotation and number of regions [5, 6, 82].

This chapter examines the effect of changing the number of region partitions and rotation of the fingerprint image to establish whether there is an ideal number of regions and a method of rotation that can produce the most unique and consistent patterns. The number of regions affects the visual noise present on a *DP* and the detection accuracy of *SP* [7, 82]. In this research, the optimal number of regions is established by evaluating the impact it has on the amount of visual noise present on the *DP* and the *SP* detection accuracy produced. Evaluating the *DP* under rotation is also conducted, since it affects the consistency and uniqueness of the *DP* of each class. Having unique patterns for each class and ensuring they are consistently achieved aids reducing inter-class variability issues.

The ideal way to analyse these factors would be to implement various classification techniques for *DPs* with different number of regions at various rotations [8]. However, most automated classification methods are designed for *DPs* that are created using a specific number of regions, and therefore cannot be implemented for multiple number of regions [7]. Alternatively, visual comparisons are made of variations in patterns, and these are analysed alongside the *SP* detection accuracy. Since the dissertation is focused on classifying flat fingerprints with missing *SPs*, the change in patterns for both complete fingerprint classes and fingerprints with missing *SPs* are observed.

The following sections establish the most appropriate number of regions to construct a *DP* and the optimal method of rotation that produces unique *DPs* for each flat fingerprint case. The first step is to form the *DP* of the flat fingerprints that are used for this investigation. This is covered in Section 3.2. The description of the experiments undertaken is discussed in Section 3.3. Section 3.4 provides the discussion of the findings obtained from experiments. Section 3.5

presents the ideal number of regions to construct a *DP* and the various methods of rotation that produces unique *DPS* for each flat fingerprint case. The conclusions are summarised in Section 3.6.

3.2 Construction of a *DP* with n regions

As part of the investigation, the construction of a *DP* for n regions is presented. The number of regions n varies, based on the requirements of the experiment conducted. To formulate a *DP* with n regions, the steps are as follows:-

1. Fingerprint pre-processing
2. Orientation field estimation
3. Region segmentation

These steps are further discussed in the sections below.

3.2.1 Pre-processing fingerprints

Before obtaining the orientation fields to formulate the *DP*, the background image needs to be discarded in order to extract only the ridges and valleys of the fingerprint. The unwanted background may be affected by noise and distortion, regarding orientation fields around the edges of the fingerprint. This leads to false *SPs* appearing around the edges of the *DP*. To eliminate possible false *SPs* or irregular *DPS*, background segmentation is performed using the method in [83].

3.2.2 Orientation Field Algorithm

The orientation fields of the segmented fingerprint are used to form the *DPS*. These fields are calculated using the least mean square orientation estimation algorithm, described in [43]. This gradient-based method is selected since it has been successfully used [60]. The steps of the algorithm are the following [43]:-

1. Normalisation
2. Ridge Orientation Estimation

3.2.2.1 Normalisation

Normalisation is applied to minimise the changes of gray level values along ridges and valleys [43], while ensuring the integrity of the data. The method used by Hong *et al.* is employed to perform normalisation [43]. The pixel at row r and column c on the fingerprint matrix, is normalised using Equation 3.1 to produce the normalised pixel $N_p(r, c)$ at location (r, c) .

$$N_p(r, c) = \begin{cases} M_0 + \sqrt{(Var_0(I(r, c) - M)^2)/(Var)} & \text{if } I(r, c) > M \\ M_0 - \sqrt{(Var_0(I(r, c) - M)^2)/(Var)} & \text{otherwise} \end{cases} \quad (3.1)$$

The estimated mean and variance is denoted by M and Var , respectively. $I(r, c)$ denotes the gray-level value at location (r, c) . The mean and variance represented by M_0 and Var_0 are assigned zero and one, respectively. The normalised pixel is stored in the normalised matrix G at location (r, c) .

3.2.2.2 Ridge orientation estimation

The normalised matrix G is used to construct the orientation field matrix $Orient(r, c)$. The summarised steps to construct the matrix are as follows:-

1. Partition G into blocks of $w \times w$, where w is 16 in this work.
2. Calculate the gradients $\partial_x(r, c)$ and $\partial_y(r, c)$ at pixel location (r, c) .
3. Using Equations 3.2 to 3.4 compute the appropriate local orientation value for each block in G , with centre (r, c) .

$$\nu_x(r, c) = \sum_{u=r-\frac{w}{2}}^{r+\frac{w}{2}} \sum_{v=c-\frac{w}{2}}^{c+\frac{w}{2}} 2\partial_x(u, v)\partial_y(u, v) \quad (3.2)$$

$$\nu_y(r, c) = \sum_{u=r-\frac{w}{2}}^{r+\frac{w}{2}} \sum_{v=c-\frac{w}{2}}^{c+\frac{w}{2}} \partial_x^2(u, v)\partial_y^2(u, v) \quad (3.3)$$

$$\theta(r, c) = \frac{1}{2} \arctan\left(\frac{\nu_y(r, c)}{\nu_x(r, c)}\right) \quad (3.4)$$

4. Apply low pass filtering to reduce noise found in $\theta(r, c)$. Φ_x and Φ_y in Equations 3.5 and 3.6 are continuous vector fields for components x and y , respectively.

$$\Phi_x(r, c) = \cos(2\theta(r, c)) \quad (3.5)$$

$$\Phi_y(r, c) = \sin(2\theta(r, c)) \quad (3.6)$$

5. Compute in $\Phi'_x(r, c)$ and $\Phi'_y(r, c)$ as:-

$$\Phi'_x(r, c) = \sum_{u=-\frac{w_\Phi}{2}}^{\frac{w_\Phi}{2}} \sum_{v=-\frac{w_\Phi}{2}}^{\frac{w_\Phi}{2}} W(u, v) \Phi_x(r - uw, c - vw) \quad (3.7)$$

$$\Phi'_y(r, c) = \sum_{u=-\frac{w_\Phi}{2}}^{\frac{w_\Phi}{2}} \sum_{v=-\frac{w_\Phi}{2}}^{\frac{w_\Phi}{2}} W(u, v) \Phi_y(r - uw, c - vw) \quad (3.8)$$

6. Compute the local ridge orientation value $O(r, c)$ at pixel (r, c) .

$$O(r, c) = \frac{1}{2} \arctan\left(\frac{\Phi'_y(r, c)}{\Phi'_x(r, c)}\right) \quad (3.9)$$

Figure 3.3 shows the axis used to calculate the orientation fields, where 0° lies on the right lower quadrant moving clockwise to 180° in the left lower quadrant. Once the orientation fields are calculated, smoothing is applied using a Gaussian filter. This reduces local orientation uncertainty.

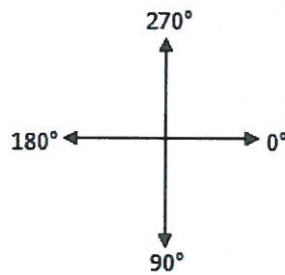


Figure 3.3: The axis used to calculate the orientation fields

3.2.3 Region segmentation for n regions

A DP is composed of different regions of orientation fields. To partition the orientation matrix $O(r, c)$ into n regions, a region segmentation method is applied. For this dissertation, a simple region segmentation method is used. The method combines all orientation fields falling within a particular range into a single region. An example of a three region segmented DP , is shown in Figure 3.4. In order for three regions to form on the DP , there must be three different orientation field ranges falling between 0° and 180° .



Figure 3.4: A DP created using three orientation ranges [7, 8]

Equations 3.10 to 3.12 are formulae used to construct a n region DP [7, 8]. Each range of orientation fields has a constant interval $\Delta\phi$ calculated using Equation 3.10, where n represents desired amount of regions. For experiment 1, the number of regions n is varied. For experiment 2, the number of regions n is a constant value that is selected based on the observations made in experiment 1.

$$\Delta\phi = 180^\circ/n \quad (3.10)$$

The range for each orientation grouping can be computed using Equation 3.11, where $i = 1 \dots n$.

$$range_i = [(i - 1) * \Delta\phi] : [i * \Delta\phi] \quad (3.11)$$

The region number that is used to identify a specific range as shown in Figure 3.4 at a particular block w with a centre (r, c) , can be derived from Equation 3.12, where $O(r, c)$ is the orientation value at location (r, c) .

$$region_{num}(r, c) = \lceil O(r, c) / \Delta\phi \rceil \quad (3.12)$$

3.3 Experimental set-up

In order to determine the optimal number of regions and optimal angle of rotation for each flat fingerprint case, two experiments were conducted. The first experiment varies the number of regions to establish which one is the most optimal number to construct a *DP*. This is determined by assessing the impact it has on the detection accuracy of *SPs* and the amount of visual noise present on the *DPs*. The second experiment varies the angle of rotation to establish which angle produces the most consistent patterns for each flat fingerprint case. Section 3.3.1 and Section 3.3.2 cover details of experiment 1 and experiment 2, respectively [8].

3.3.1 Experiment 1: Evaluating the impact of varying the number of regions

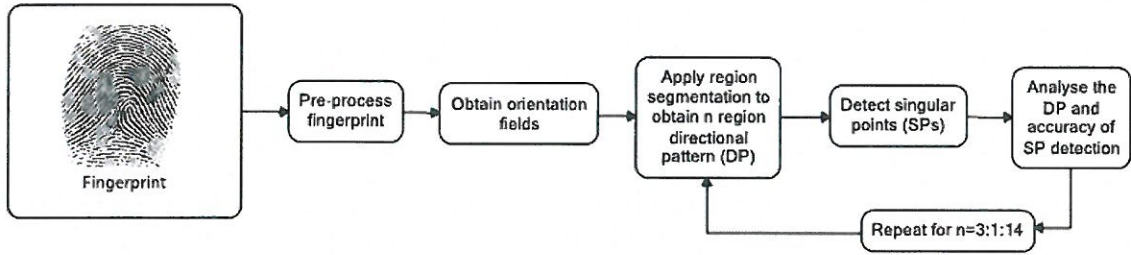


Figure 3.5: Flow diagram of the experimental set-up used to evaluate the impact of varying the number of regions, on the *SP* detection accuracy and the amount of visible noise produced [8]

Figure 3.5 depicts the overview of experiment 1 which analyses the amount of visible noise present on a *DP* and the result of the *SP* detection accuracy for each number of region. Each individual fingerprint undergoes pre-processing to remove the background image as described in Section 3.2.1. The segmented fingerprint is then represented as an orientation matrix O as described in Section 3.2.2. The region segmentation is then performed on the orientation matrix to obtain the n region *DP* of the segmented fingerprint as described in Section 3.2.3. The region number n is varied from three to 14.

The accuracy of *SP* detection and amount of visible noise for each n region pattern is observed. The method for the detection of *SPs* and the observation made on the amount of visible noise present in the *DPs* are covered in Section 3.3.1.1 and Section 3.3.1.2 respectively.

3.3.1.1 Singular point detection

The point where all regions converge on a DP , is a SP [7]. However, all regions may not intersect at a single point. On that account, a neighbourhood (ND) of 24 or 48 pixels is searched for a possible SP . In this dissertation a 24 and 48 pixel search is referred to as 24 ND and 48 ND , respectively [8]. Figure 3.6 (a) depicts an example where a 24 ND (five \times five matrix) is used to detect a SP at location (r, c) , on a three region DP .

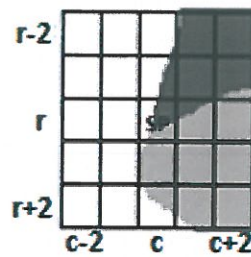


Figure 3.6: An example illustrating a SP where all three region intersect using a 24 ND [7]

There are two types of SP s, namely: loops, and deltas. The direction in which regions flow from region number 1 to n , indicates the type of SP [7]. Regions which flow in an anti-clockwise direction from region number 1 to n , indicate a loop. Conversely, regions that flow in a clockwise direction from region number one to n , indicate a delta.

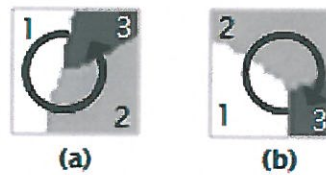


Figure 3.7: Type of SP s (a) loop and (b) delta [7]

Figure 3.7 (a) and Figure 3.6 (b) depict a loop and delta, respectively. For this experiment, the coordinates of the detected loops and deltas on a DP are compared to its ground truth results to determine the detection accuracy of SP s. The ground truth results contain the manually detected coordinates of all loops and deltas found on each fingerprint used for this experiment.

3.3.1.2 Observation made on the DP

The following observations are made for experiment 1:-

1. The detection accuracy of *SPs* versus n number of regions.
2. The amount of visible noise present for each n region *DP*.

3.3.2 Experiment 2: Evaluating the effect of rotation on DPs

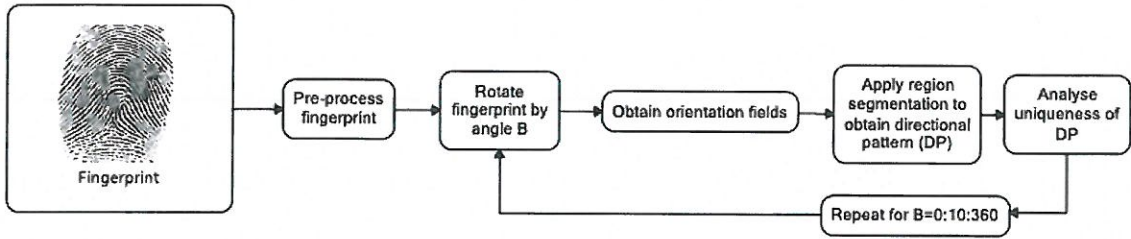


Figure 3.8: Flow diagram of the experimental set-up that evaluates the effect of varying the angle of rotation

Figure 3.8 depicts the overview of experiment 2 which analyses *DPs* produced by fingerprints that are rotated at different angles [8]. To conduct the experiment, the fingerprint is pre-processed and rotated by an angle B , from 0° to 360° at increments of 10° . Once each image is rotated, the orientation fields are obtained as described in Section 3.2.2. Region segmentation is applied on the orientation matrix O as described in Section 3.2.3. The number of regions n is set based on the outcome in Section 3.3.1. The uniqueness and consistency of the *DP* is then observed.

3.3.2.1 Increments for rotation

In this experiment, each fingerprint contained in the testing *DB* are rotated by an angle B . The angle is incremented by 10° , from 0° to 360° . A 10° increment allows for the observation of subtle changes of the *DP*.

3.3.2.2 Observation made on the DP

As the angle of rotation is varied, the following observations are made:-

1. Uniqueness of *DP* for each class containing a certain number of loops and deltas.

2. Number of layouts for a specific *DP* of each class containing a certain number of loops and deltas.
3. Consistency of the layouts.

3.3.3 Testing Samples

Fingerprints from the Fingerprint Verification Competition (*FVC*) 2002 *DB* 1a [57] were chosen for testing experiment 1. Since visual comparisons are made on the pattern variations in experiment 2, a smaller testing sample is used. For testing experiment 2, 104 upright fingerprints from the *FVC* 2002 *DB* 1a is used [8]. Due to a shortage of single delta images, 100 test images for single delta cases were created by cropping *RLs* and *LLs* from the *FVC* 2002 *DB* 1.

3.4 Discussion

3.4.1 Observations made on the *DPs* with different the number of regions

The *SP* detection accuracy and amount of visible noise present on *DPs* are observed, to determine the optimal number of regions.

3.4.1.1 Impact of the varying the number of regions on the *SP* detection accuracy

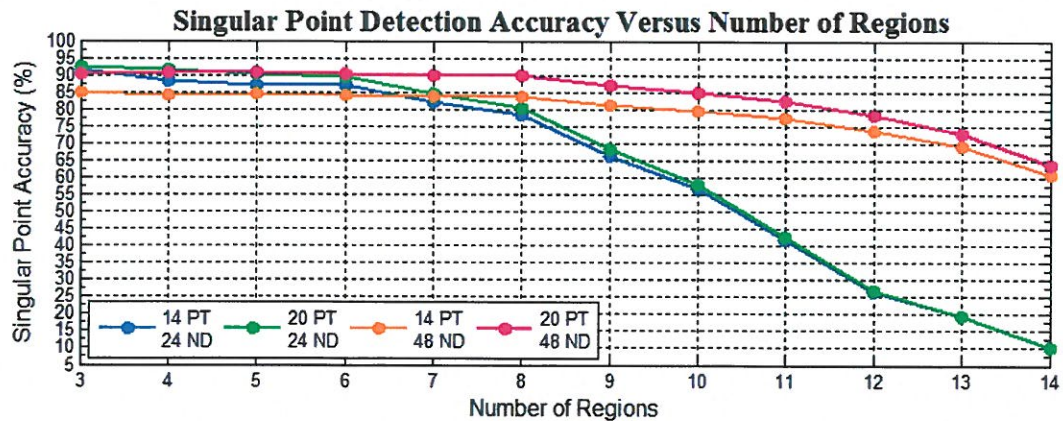


Figure 3.9: *SPs* detection versus number of regions (n) by varying the value of the neighbourhood (ND) and the pixel tolerance (PT) [8]

Figure 3.9 depicts the overall *SP* detection accuracy achieved for n region *DP*. Four experimental tests were conducted using different pixel tolerance (*PT*) and *ND* values. In this dissertation, the *PT* value is defined as the total amount of pixels that the detected loop and delta coordinates can be from its ground truth value, in order for the detection to be considered correct.

The conditions that need to be satisfied in order for the detection to be correct are as follows:

1. The type of *SP* has to be correct.
2. The number of pixels from the automatically detected loops and deltas to the ground truth locations, must be less than the *PT* value.
3. No false *SPs* must appear in the particular image.
4. All true *SPs* have to be detected.

When selecting the *NDs* for the experiment, the matrix needs to have the same number of pixels on each side (either horizontally or vertically) from a central point. To achieve this condition, $p \times p$ matrix is used (as opposed to $p \times m$) and p has to be set to an odd number. An 8 *ND* is a 3×3 matrix that is the smallest possible *ND* that has the same number of pixels horizontally and vertically from the central pixel point. Since it consist of a total of 8 pixels around a single pixel, it is less than 14 (the maximum number of regions under observation) and hence not suitable for this experiment. The next possible *ND* is 24 which is a 5×5 matrix. It has a total of 24 pixels around a central pixel point. This is greater than 14 and therefore suitable for testing purposes. To truly observe the effect of *DPs* with large number of regions on *SPs*, a larger *ND* value needs to be chosen as well for this experiment. Since specks of noise (isolated regions) can appear on the *DP* and the *SP* detection identifies the different n pixel values around a central point, there is a possibility that the isolated region can be detected as one of the regions that is attached to the true *SP*. This can compromise the results since false *SPs* can be detected. To prevent this from occurring, the value of the *ND* needs to be slightly bigger than a 24 *ND*. To satisfy the condition when selecting a *ND* (i.e., it must be a $p \times p$ matrix where p is an odd number), the next choice from a 5×5 matrix is a 7×7 matrix which is a *ND* of 48. This *ND* is double the size of a 24 *ND* and has enough pixels around a central pixel point to observe the effect of multiple intersecting points.

Table 3.1 depicts the *PT* and *ND* values selected to conduct the four tests shown in Figure 3.9. It also contains the reasoning, for selecting a particular value.

Table 3.1: PT and ND values selected for testing purposes [8]

PT Value	Reason for selecting PT value	ND value	Reason for selecting ND Value
14	Accommodates average amount of error, in automated detection.	24	Smallest ND that can realistically detect all 14 regions.
20	Used to observe the difference if a more lenient criterion is used.	48	Used to observe the effect of multiple intersecting points.

From the graph illustrated in Figure 3.9, the results show that as the number of regions increased the SP detection accuracy for each test dropped. Analysing the area around the converging point of DP s with larger number of regions will most likely provide the reasoning for the low SP detection accuracy achieved for larger number of regions.

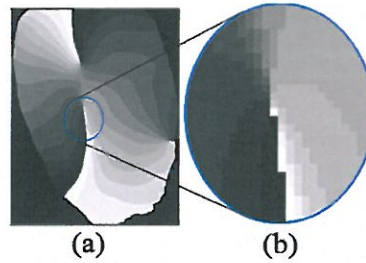


Figure 3.10: A W class represented by a (a) 15 region DP , and (b) zoomed in at its converging region, which is not a SP [8]

A W fingerprint DP with a large number regions of 15, is shown in Figure 3.10. The converging region is zoomed in for observation. From the observation, all regions did not intersect at a single point. For a SP to be located, all 15 regions need to be detected. However, there are multiple points of intersection of three and four regions. The number of multiple intersecting points of three and four regions seems to increase, as the number of regions on a DP increases [8]. Therefore, a gradual drop in accuracy for all tests occur, even when searching a ND rather than a single point of intersection.

However, some tests or conditions experience a larger drop in accuracy than others. This is owing to the fact that certain conditions can alleviate the problem of multiple intersecting points of three and four regions, that contribute towards the drop in accuracy. These conditions are as follows:-

1. An increase in the ND value from 24 to 48.
2. An increase in the PT value from 14 to 20.

The tests conducted using a 48 ND , produce better results than a 24 ND for 6 to 14 regions, which have the same PT value. This is observed from the graph in Figure 3.9. The higher ND value is large enough to detect all n regions for larger number region. Additionally, the tests conducted using a 20 PT , produced slightly higher results than a 14 PT for six to 14 regions, when compared against tests that have the same ND value.

However, the number of regions which produced the highest accuracy for all tests conducted, regardless of the ND and PT value used, was a three region DP . From the graph, three regions produced a high accuracy of 92.1%, using a 20 PT value and a 24 ND value. Based on these findings, three is the optimal number of regions that a DP should have.

3.4.1.2 Impact of varying the number of regions on the amount of visible noise present on DPs

To conduct this test, the smoothing on orientation fields that would of been applied in the pre-processing stage will not be performed. This will allow visible noise on the DP to appear, as the number of regions on the fingerprint DP is varied. Visible noise appears as isolated regions on the DP .

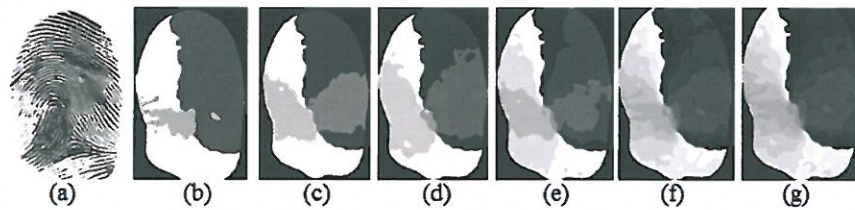


Figure 3.11: (a) Noisy fingerprint image with DP s formed using different numbers of regions namely: (b) 3, (c) 4, (d) 5, (e) 7, (f) 12 and (g) 15 [8]

Examples shown in Figure 3.11 (b) to Figure 3.11 (g) illustrates the effect of increasing the number of regions (n), on the amount of visible noise present on the DP . These fingerprint DP s represent a LL with smudges and missing ridges shown in Figure 3.11 (a). Table 3.2 provides the observations made on the n region DP s which are depicted in Figure 3.11.

Smaller number of regions like three or four, have less visible noise appearing on the DP . Since the size of the regions are much larger, noise appears as isolated regions. This makes it easier to identify and remove the noise.

Table 3.2: Summary of the observations made on the *DP* with n regions [8]

Smaller number of regions (3 to 6)	Larger number of regions (> 6)
Less visible noise	More visible noise
Easier to identify and remove. Noise represented as isolated regions since region size is large.	Difficult to identify noise, where the region size is smaller. Appears as one of the <i>DP</i> regions.
<i>DP</i> is distinct	<i>DP</i> is distorted

3.4.2 Variation of rotation

In this section, the pattern changes were observed as the flat fingerprints were rotated. Each flat fingerprint class containing the same number of captured loops and deltas was observed simultaneously. Classes with the same number of loops and deltas suffer from small inter-class variability. Hence, observing these classes simultaneously makes it easier to identify unique patterns on which the classification rules will be based on. In this dissertation the layout of any class is identified by the number of common regions (*CRs*). These *CRs* are regions that are common between all the loops and deltas found on the *DP*.

3.4.2.1 Observation of the *W* DP with two loops and two deltas, under rotation

Amongst all classes, *Ws* are the only class that has two loops and two deltas. As a result, a *W* can be classified immediately by detecting two loops and/or two deltas at any rotation [8]. For completeness, *Ws* were observed under rotation. Figure 3.12 shows the *DPs* of a *W* fingerprint that is rotated at angles, (b) 0° , (c) 10° , (d) 30° , (e) 50° and (f) 90° . For a *W* class with two loops and two deltas, the structure between the top and bottom loops is symmetrical as depicted in Figure 3.12. All the orientation fields that represent the ridges between the loops and deltas, fall within the same range and therefore have the same region number. Hence, a single *CR* forms between all *SPs* as seen in Figure 3.12 (b). As a result, the layout is also unique. Furthermore, it was observed that the region number of the largest *CR* only changes to the next region number after the fingerprint has been rotated approximately 50° . For example, from Figure 3.12 (b) to (c) the region number of *CR* is 2 and thereafter it changes to 3.

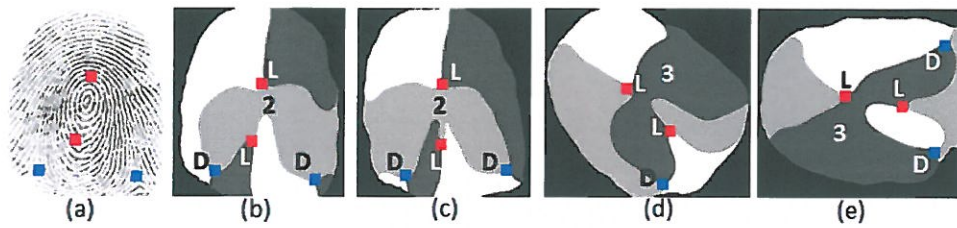


Figure 3.12: The DPs of an upright W that is rotated at angle, (b) 0° , (c) 10° , (d) 50° and (e) 90° [8]

3.4.2.2 Observation of the W , RL , LL and TA DPs with one loop and one delta, under rotation

The possible classes for a flat fingerprint with a single loop and single delta captured, are W s, RL s, LL s and TA s. Owing to the structural layout of a W , there are two types of loops: a top loop and a bottom loop as shown in Figure 3.13. An incomplete W may have a bottom loop and a delta, or a top loop and a delta. Since a bottom loop lies between a top loop and delta, it is very unlikely to have a case of a W with a top loop and a delta. Therefore, these cases will not be investigated. However, a bottom loop and a delta is more likely to occur. Identifying a bottom loop immediately suggest that the class is a W , therefore the difference between a bottom loop and top loop is analysed.



Figure 3.13: Indication of a top loop and a bottom for W fingerprint image

The difference between the bottom loop and top loop is the flow of orientation fields towards the loops and the position of loops on the fingerprint. The arrows in Figure 3.13 indicate the flow of the orientation fields. For a top loop, orientation fields flow away from the loop. For a bottom loop the orientation fields flow towards the loop. Furthermore, the orientation fields move in the

same direction between the top loop and bottom loop and therefore are represented by a single region as seen in Figure 3.14 (c).

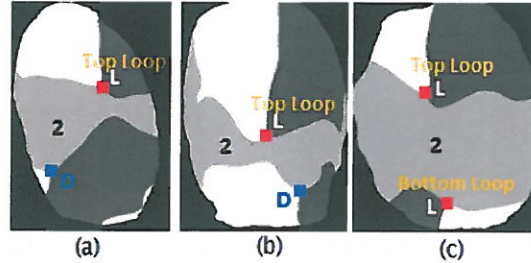


Figure 3.14: Indication of a top loop for (a) *RL DP*, (c) *LL DP*, and both a top loop and bottom loop on (d) a *W DP*

Based on the findings in Section 3.4.2.1 the *CR* which is represented by region 2 lies above the bottom loop and changes to the next region number after 50° . For *RLs*, *LLs* and *TAs* it was found that region 2 lies below the top loop as depicted in Figure 3.14. Therefore by detecting region 2 relative to the loop, the bottom loop can be differentiated from a top loop. These cases will not be observed under rotation since the maximum rotation of a fingerprint in the *FVC DB* (*DB* used for testing) does not exceed 35° and region 2 changes only after 50° [57].

Unlike *Ws*, classes like *RLs*, *LLs* and *TAs* have to be observed under rotation to identify its uniqueness. Owing to the number of loops and deltas, *RLs*, *LLs* and *TAs* were observed simultaneously. It was found that more than one layout can be produced at the same angle owing to the amount of ridges flowing in the same direction of the loop for an *RL*, *LL* and *TA*. A layout is identified by the number of *CRs* between the *SPs*.

RL, *LL* and *TA* fingerprints form *DPs* that have three different types of layouts. Figures 3.15 and 3.16 depict these three different layouts, namely: 3-*CR*, 2-*CR* and 1-*CR DP* layout produced by *RLs* and *LLs*, respectively. Conversely, for a *TA* only two layouts are produced, namely: 3-*CRs* and 2-*CRs*. Figure 3.17 shows the different layouts of a *TA*.

Figures 3.18 to 3.24 depicts a great variation of layouts under rotation, regardless if the fingerprint formed a specific type of layout at its upright position. Therefore, the inconsistency of patterns is seen at each angle of rotation.

An example of this, is illustrated in Figure 3.18 (g) and 3.18 (h), where a *RL* fingerprint rotated by 160° produces a 2-*CR DP* that changes to 3-*CR* when the fingerprint is rotated by 180° .

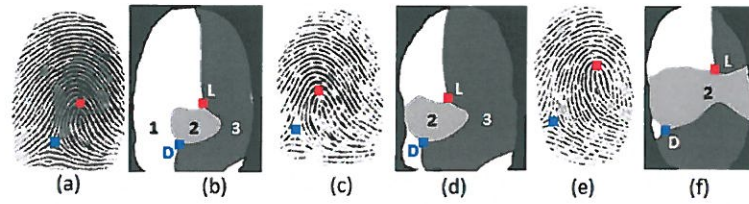


Figure 3.15: Three different types of *DP* layouts produced from (a) an *RL* fingerprint that forms a 3-*CR* layout, (b) a 3-*CR* layout, (c) an *RL* fingerprint that forms a 2-*CR* layout, (d) a 2-*CR* layout, (e) an *RL* fingerprint that forms a 1-*CR* layout and (f) a 1-*CR* layout [8]

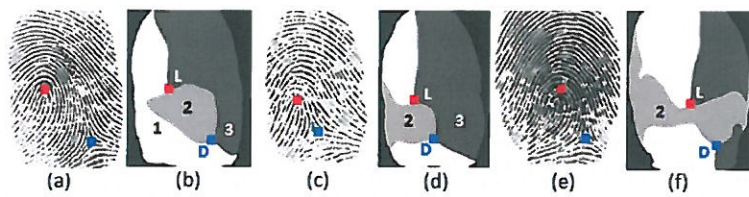


Figure 3.16: Three different types of *DP* layouts produced from (a) an *LL* fingerprint that forms a 3-*CR* layout, (b) a 3-*CR* layout, (c) an *LL* fingerprint that forms a 2-*CR* layout, (d) a 2-*CR* layout, (e) an *LL* fingerprint that forms a 1-*CR* layout and (f) a 1-*CR* layout [8]

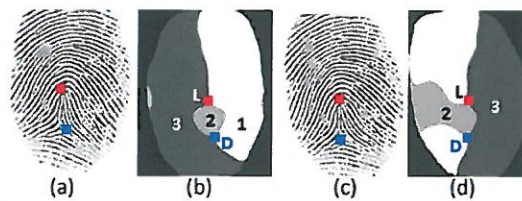


Figure 3.17: Two different types of *DP* layouts produced from (a) a *TA* fingerprint that forms a 3-*CR* layout, (b) a 3-*CR* layout, (c) a *TA* fingerprint that forms a 2-*CR* layout and (d) a 2-*CR* layout [8]

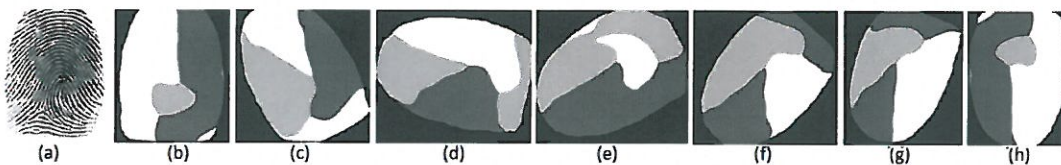


Figure 3.18: The *DPs* formed when an *RL* fingerprint that produces a 3-*CR* layout at its upright position is rotated at angles, (b) 0°, (c) 30°, (d) 80°, (e) 120°, (f) 140°, (g) 160° and (h) 180° [8]

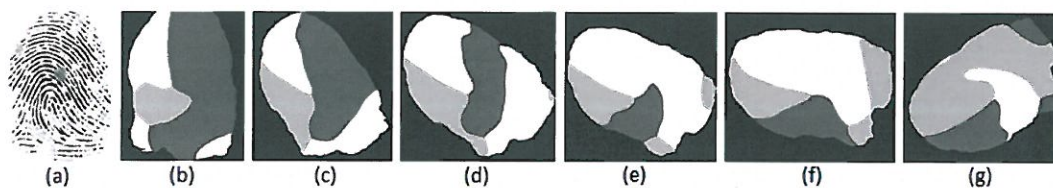


Figure 3.19: The *DPs* formed when an *RL* fingerprint that produces a 2-*CR* layout at its upright position is rotated at angles, (b) 0°, (c) 20°, (d) 50°, (e) 60°, (f) 80° and (g) 120° [8]

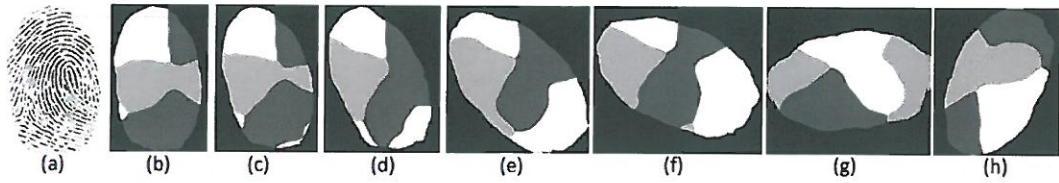


Figure 3.20: The *DPs* formed when an *RL* fingerprint that produces a 1-*CR* layout at it's upright position is rotated at angles, (b) 0°, (c) 10°, (d) 20°, (e) 30°, (f) 60°, (g) 90° and (h) 150° [8]

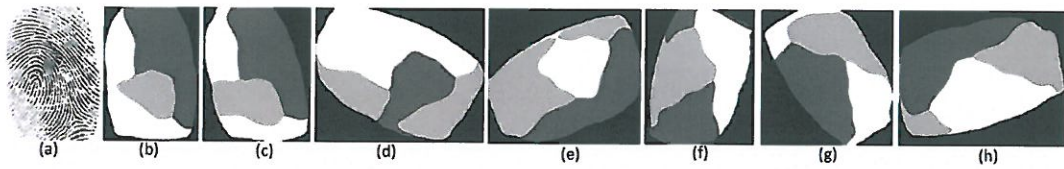


Figure 3.21: The *DPs* formed when an *LL* fingerprint that produces a 3-*CR* layout at it's upright position is rotated at angles, (b) 0°, (c) 10°, (d) 60°, (e) 110°, (f) 160°, (g) 210°, and (h) 280° [8]

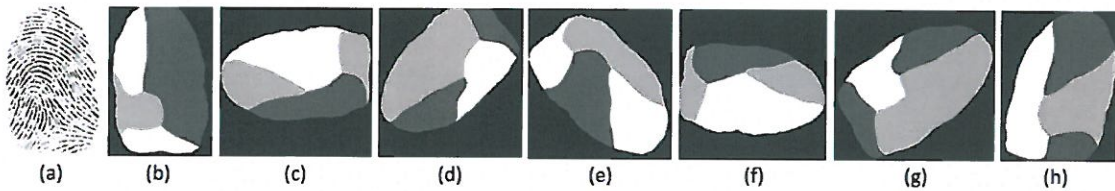


Figure 3.22: The *DPs* formed when an *LL* fingerprint that produces a 2-*CR* layout at it's upright position is rotated at angles, (b) 0°, (c) 90°, (d) 130°, (e) 220°, (f) 260°, (g) 300° and (h) 340° [8]

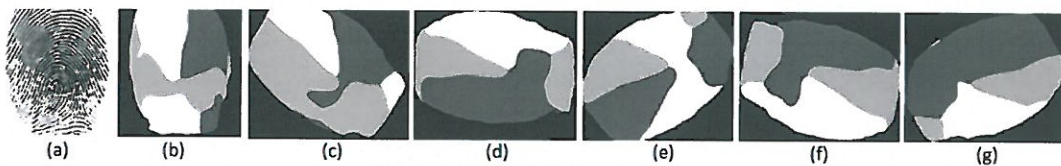


Figure 3.23: The *DPs* formed when an *LL* fingerprint that produces a 1-*CR* layout at it's upright position is rotated at angles, (b) 10°, (c) 60°, (d) 90°, (e) 130°, (f) 260°, and (g) 300° [8]

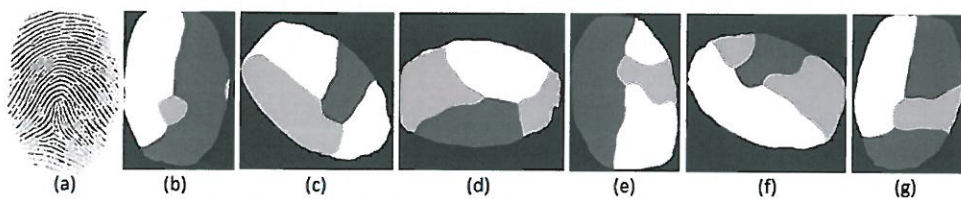


Figure 3.24: The *DPs* formed when a *TA* fingerprint that produces a 3-*CR* layout at it's upright position is rotated at angles, (b) 10°, (c) 50°, (d) 90°, (e) 190°, (f) 250° and (g) 350° [8]

These pattern inconsistencies are also seen for *LLs*, in Figures 3.21 to 3.23; and for *TAs*, in Figure 3.24.

It is therefore clear that *RL*, *LL* and *TA* classes produce inconsistent patterns under rotation. In addition, even at the same rotation the patterns are not unique for each class.

From the example shown in Figure 3.25 to 3.27, the *DPs* of the same layout which are formed from the different classes are highly similar. These similarities make it challenging to overcome small inter-class variabilities between the fingerprint classes. The only major visual difference between classes that produce the same *DP* layout, is the position of the loop relative to the delta. Basing a classification only on the position of *SPs*, will imply that the fingerprints have to be upright. However, this is never the case as captured fingerprints experience rotation.

Since the patterns vary under rotation, a consistent method of rotating a fingerprint, regardless of its initial rotation angle, needs to be established. In addition, the rotation must produce unique patterns for each class, which is not the case at most standard rotations as shown above.

After analysing the *DP* under rotation, a method of rotation was established. Aligning the *SPs* either vertically or horizontally satisfied the key requirements that meets the objective of consistency and uniqueness. The *DPs* then became unique for each class. Furthermore, using fingerprint landmarks instead of a global angle ensure the rotation is consistent. This is depicted by Figure 3.28, in which *SPs* are vertically aligned.

Figures 3.28 (a) to 3.28 (c) depicts unique 2-*CR DP* layouts formed from an *LL*, *RL* and *TA* fingerprint. The *CR* region numbers for each of these classes differ. An *LL*, *RL* and *TA* have *CR* region numbers of: 1 and 2; 3 and 2; and 2 and 3, respectively. Figures 3.28 (d) to 3.28 (f) shows unique 3-*CR DP* layouts formed from an *LL*, *RL* and *TA*. The smallest *CR* region number for each of these classes differ. The smallest *CR* region number for an *LL*, *RL* and *TA* is: 1; 3; and 2, respectively. If this method of rotation is applied in fingerprint classification, it will not be dependent on the location of the loop and delta to classify a fingerprint but rather on the unique patterns produced. On that account, a proposed technique to obtain consistency and unique class patterns, is to align the *SPs*.

The reason that the region numbers of the *CRs* are unique to an *RL*, *LL* and *TA*, is that the loop flow varies for the each class. The location of the delta relative to the loop also differ for each class [10]. The delta lies on the left and right for an *RL* and *LL*, respectively. Therefore, aligning the loop and delta, rotates an upright *RL* and *LL* to the right and left, respectively. Conversely, for an upright *TA* the loop lies below the delta, therefore it is rotated only slightly.

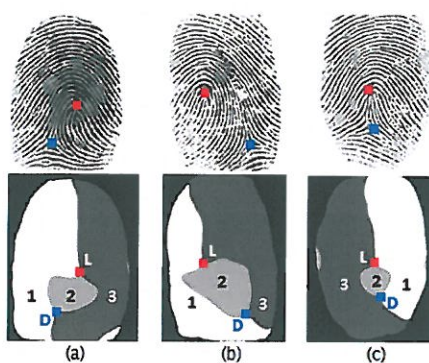


Figure 3.25: 3-CR layouts at an angle of 0° that was formed by classes (a) *RL*, (b) *LL*, and (c) *TA* [8]

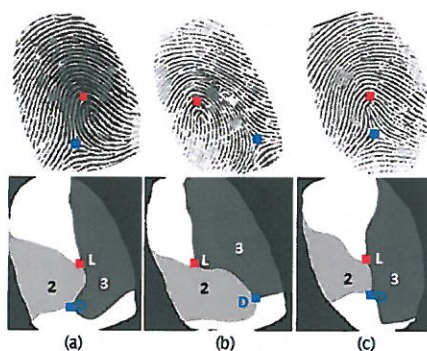


Figure 3.26: 2-CR layouts at an angle of 10° that was formed by classes (a) *RL*, (b) *LL*, and (c) *TA* [8]

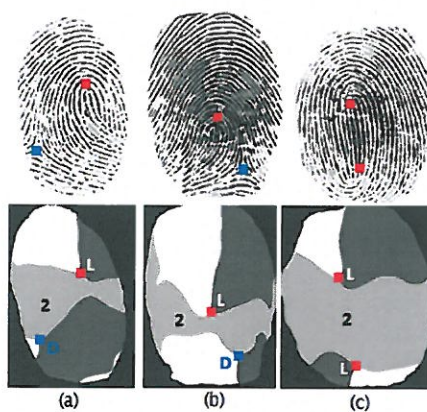


Figure 3.27: 1-CR layouts at an angle of 0° that was formed by classes (a) *RL*, (b) *LL*, and (c) *W* [8]

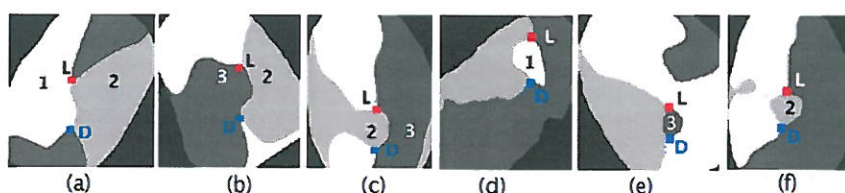


Figure 3.28: Fingerprints which are rotated such that the *SPs* are vertically aligned produces a 2-CR layout of class, (a) *LL*, (b) *RL*, and (c) *TA*; and a 3-CR layout of class (d) *LL*, (e) *RL*, and (f) *TA* [8]

These differences in these rotation create unique *DPs*.

3.4.2.3 Observation of *RL*, *LL*, *TA* and *W* *DPs* with one loop under rotation

The possible classes for a flat fingerprint with a single loop captured are *Ws*, *RLs*, *LLs* and *TAs*. *Ws* can have either a single bottom loop or a single top loop. The observation of a bottom loop versus a top loop was covered in Section 3.4.2.2. Hence it will not be addressed again herein.

An *RL*, *LL*, *TA* and *W* flat fingerprint with only one top loop captured will be observed simultaneously. Figure 3.29 to Figure 3.32 illustrate the *DPs* that are formed when the fingerprints were rotated from 0° to 360° .

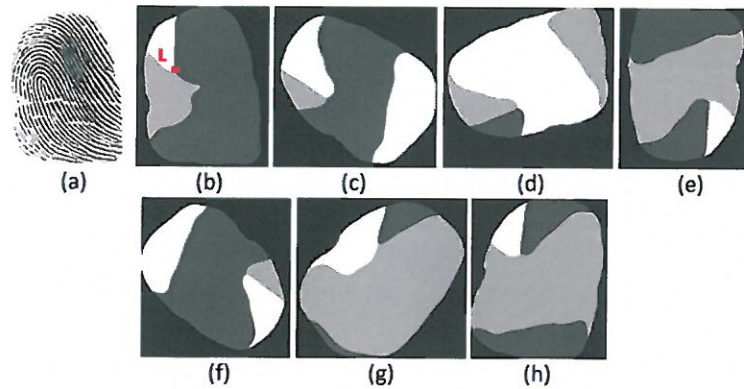


Figure 3.29: The *DPs* formed from an *RL* with a single loop that is rotated at angles of, (b) 0° , (c) 50° , (d) 100° , (e) 170° , (f) 220° , (g) 310° and (h) 340° [8]

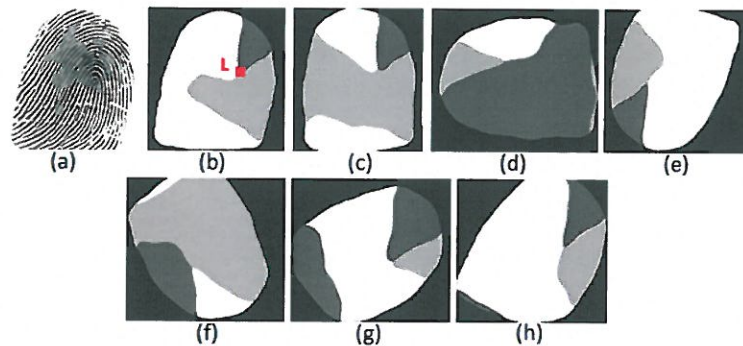


Figure 3.30: The *DPs* formed from an *LL* with a single loop that is rotated at angles of, (b) 0° , (c) 10° , (d) 100° , (e) 170° , (f) 220° , (g) 310° and (h) 340° [8]

The *DPs* for the *RL*, *LL*, *TA* and *W* class with only one loop captured are inconsistent for each angle. Conversely, it was found that when the images are rotated so that the loop direction for

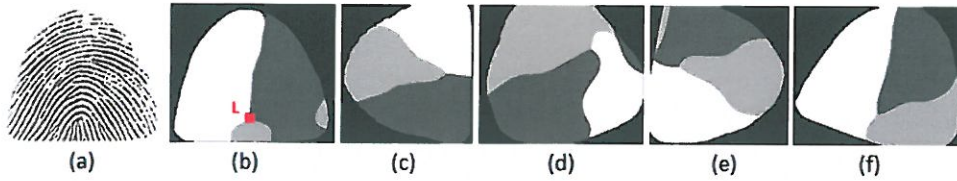


Figure 3.31: The *DPs* formed from a *TA* with a single loop that is rotated at angles of (b) 0° , (c) 90° , (d) 130° , (e) 260° , and (f) 340°

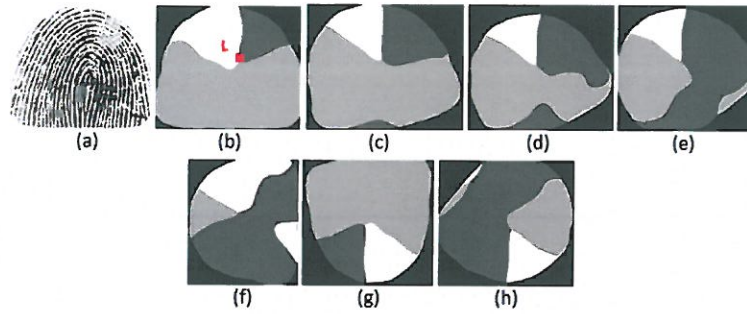


Figure 3.32: The *DPs* formed from a *W* with a single loop that is rotated at angles of, (b) 0° , (c) 10° , (d) 20° , (e) 40° , (f) 90° , (g) 190° and (h) 230° [8]

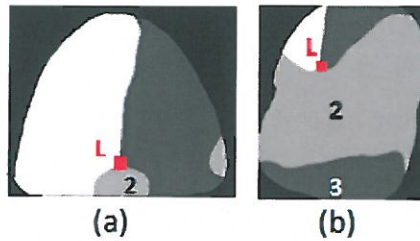


Figure 3.33: Region 2 that links to the loop *L* found on (a) a *TA* does not extend to the side of the *DP*, whereas for (b) an *RL* it extends to the sides of the fingerprint

all three classes was pointing downwards, unique patterns appeared. *TA* is the only class where region 2 (that is connected to the loop) does not extend to the sides of the fingerprint. However, for *LLs*, *RLs* and *Ws*, region 2 (that is connected to the loop) touches the sides of the fingerprint as depicted in Figure 3.33. Region 2 is also found to be the largest region connected to the loop.

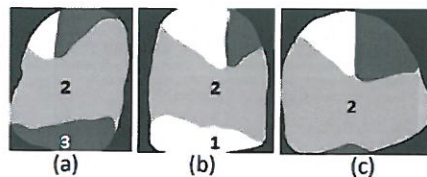


Figure 3.34: The *DPs* which are produced by classes that have a single loop, namely: (a) *RL*, (b) *LL* and (c) *W* [8]

In addition, it was discovered that the regions below region number 2 are unique for an *RL*, *LL*

and W , as seen in Figure 3.34. The region numbers represent the flow of the ridges as it enters or exits the fingerprint. The flow of the ridges of an RL and LL advances to the right and left, respectively resulting in region numbers below region 2 to differ. Conversely, for a W the flow of ridges converge as it approaches an intersecting point which would of formed the bottom loop that has not been captured. This results in region numbers below region 2 to be 3 and 1. The flow of ridges for an RL , LL and W is indicated by arrows in Figure 3.35.



Figure 3.35: The flow of the orientation fields at the bottom of the fingerprint differ for class (a) RL , (b) LL and (c) W when the loop direction of each of these classes points downwards

3.4.2.4 Observation of the RL and LL DPs with one delta, under rotation

The main information about a class lies below the loop and above the delta. When a majority of this area is excluded, it is often difficult to identify the class. Therefore, a single delta fingerprint under rotation has to be closely analysed to establish a unique pattern.

The possible classes for a flat fingerprint with a single delta are RL s, LL s, W s and TAs . No uniqueness was found for a W or a TA with a single delta and therefore these have not been investigated under rotation. The observations are made only on RL s and LL s with a single delta.

Figure 3.36 and Figure 3.37 show the DP produced by a single delta fingerprint under rotation for an RL and LL .

It was found that unique patterns for each class appeared in some cases, for example at a rotation of 0° , 80° , 120° , 200° , and 350° . The largest region of each class has a unique region number. However, this is dependent on the location of the delta and the amount of information captured on either side of the delta.

In an attempt to find an alternate unique pattern, a fingerprint that has the same length on either side of the delta was observed under rotation. The DP results for this fingerprint are shown in Figure 3.38. It was found that the sizes of regions are fairly similar. On that account, it is unreliable to base the classification on the side in which the largest region lies.

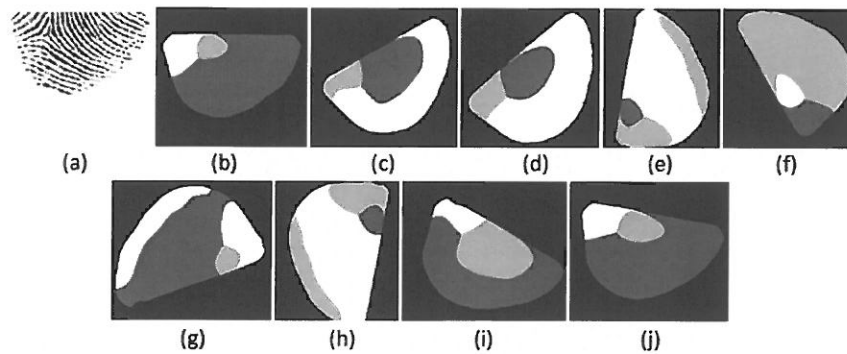


Figure 3.36: The different patterns formed from an *RL* with a single delta that is rotated at angles of, (b) 0°, (c) 30°, (d) 40°, (e) 80°, (f) 120°, (g) 200°, (h) 260°, (i) 330°, and (j) 350°

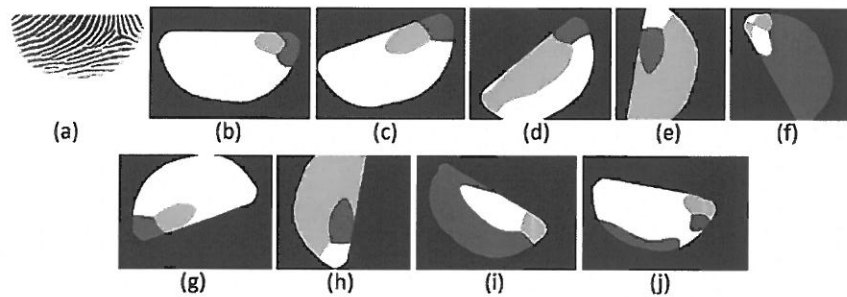


Figure 3.37: The different patterns formed from an *LL* with a single delta that is rotated at angles of, (b) 0°, (c) 30°, (d) 40°, (e) 80°, (f) 120°, (g) 200°, (h) 260°, (i) 330°, and (j) 350°

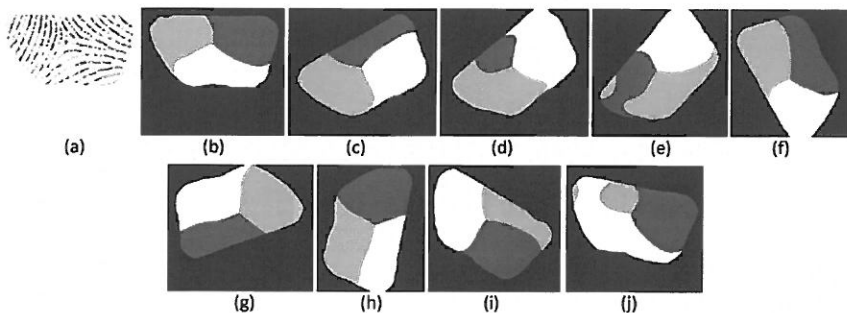


Figure 3.38: The different patterns formed from a rotated *RL* with a single delta that has the same length on either side of the delta. It is rotated at angles of, (b) 0°, (c) 30°, (d) 40°, (e) 80°, (f) 120°, (g) 200°, (h) 260°, (i) 330°, and (j) 350°

A method of consistent rotation similar to previous cases is an alternative. A unique pattern was found when the image was rotated such that the orientation fields below the delta were horizontal, as shown in Figure 3.39.

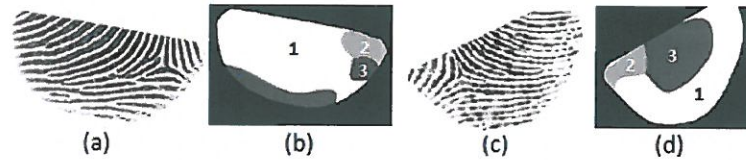


Figure 3.39: Class (a) *LL* fingerprint with a single delta, that produces (b) a unique *LL DP*, and class (c) *RL* fingerprint with a single delta, that produces (d) a unique *RL DP*

Region 2 is positioned on different sides of the delta, for *RL* and *LL* classes. This region represents the ridges found just above the delta that lie on the opposite side of the flow of the loop. Since the flow of the loop relative to the delta differs for *RL* and *LL*, the position of region 2 is unique.

3.4.2.5 Observation of the *PA* and *PF DP* with no *SPs* under rotation

The *PA* class has a unique ridge structure. It is the only complete class that has no *SPs*. However, some fingerprints are cut off so much that the fingerprint class pattern is not captured. This results in partial fingerprints (*PFs*) that also contain no *SPs*. Since these classes both have no *SPs*, most classification techniques ignore *PF* cases and immediately classify them as *PA*s. However, a *PF* could be a *W*, *RL*, *LL*, *TA* or even a *PA*. Therefore, a classification rule needs to be created for a *PA* and a *PF*. To achieve this the classes need to be observed under rotation to determine their unique characteristics.

Individuals often place the tips of their fingers on the scanner. As a result of this, the most common *PF* case occurs when the fingerprint is cut off just above the loop (point where the highest ridge curvature occurs). In this dissertation, only the most common *PF* case is addressed. Since these cases have lost most of the class ridge information, not even a human can not identify the type of class. Hence, it can not be further categorised as an *RL*, *LL*, *TA*, *W* or a *PA*.

The *PA* and *PF* classes are observed under rotation. Figure 3.40 and 3.41 shows the *DP* results for a *PA* and a *PF* fingerprint under rotation, respectively.

From the observation made (for both *PA* and *PF*), the regions never intersect at any rotation since there are no *SPs*. However, the highest ridge curvature point can be located. The area around this point will have the most information about a *PA* and a *PF*. The highest curvature

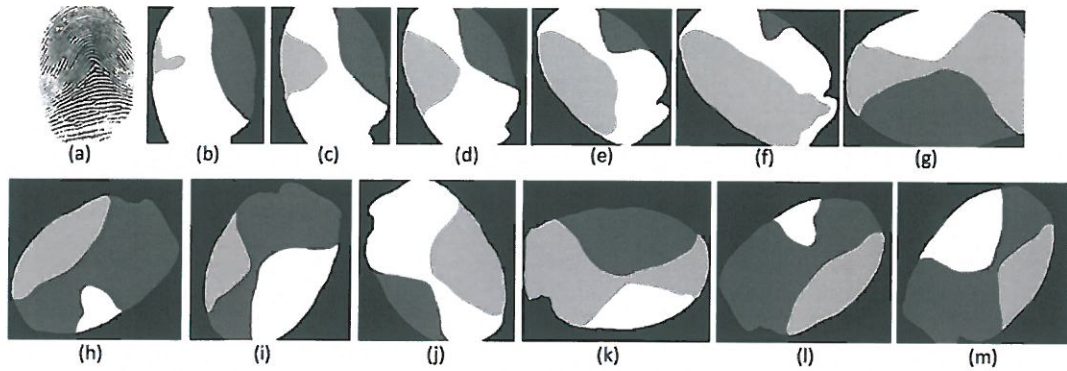


Figure 3.40: The different patterns formed from a rotated *PA* at angles of, (b) 0°, (c) 20°, (d) 30°, (e) 40°, (f) 60°, (g) 90°, (h) 120°, (i) 150°, (j) 210°, (k) 270°, (l) 300°, and (m) 320°

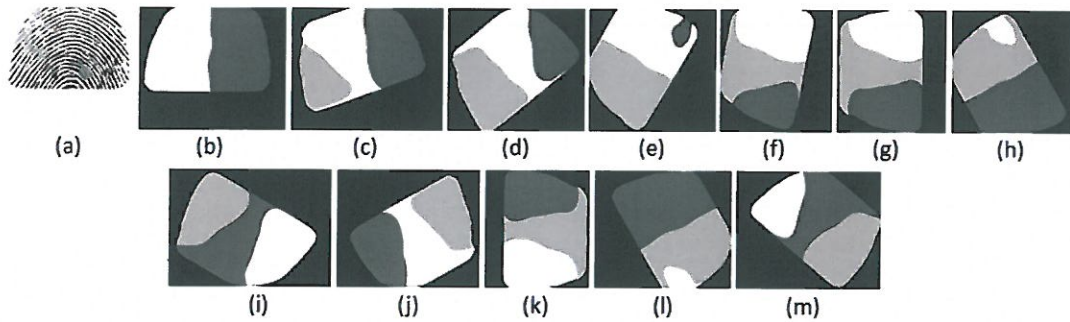


Figure 3.41: The different patterns formed from a rotated *PF* at angles of, (b) 0°, (c) 20°, (d) 30°, (e) 40°, (f) 60°, (g) 90°, (h) 120°, (i) 150°, (j) 210°, (k) 270°, (l) 300°, and (m) 320°

lies at different points for a *PA* and a *PF*, as shown in Figure 3.43. At any rotation, the highest curvature for a *PA* lie in the middle area of the class pattern. Whereas for a *PF* it lies at the edge of the fingerprint, since it is cut-off just before the intersecting point. The highest curvature on a fingerprint is represented on the *DP* by the area where all regions converge.

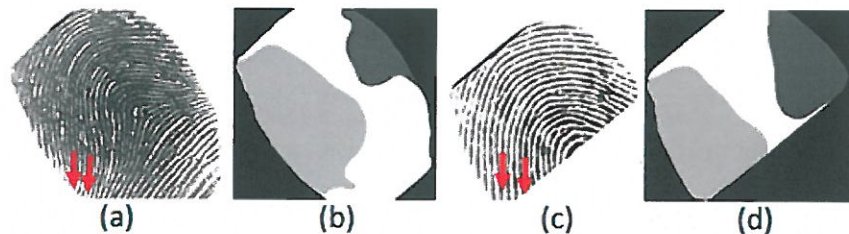


Figure 3.42: Unique *DP* for (a) a *PA* and (b) a *PF*

Figure 3.42 shows that this characteristic is more clearly seen when an upright fingerprint is rotated such that the orientation fields lying at the left side of the fingerprint points downwards.

It makes the point where all regions converge more pronounced. An example is shown in Figure 3.40 (d) to (e). Whereas, in Figure 3.41 (b) the area where region converge is not as pronounced.

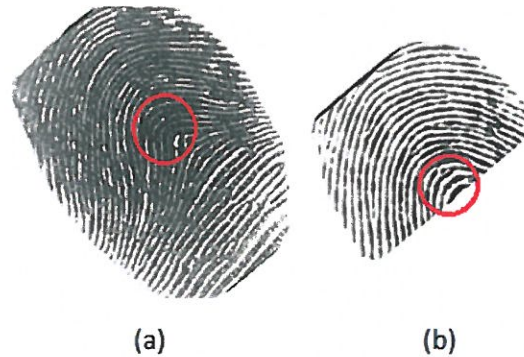


Figure 3.43: Highest curvature point within the circle for (a) *PA* and (b) *PF*

If the area where all regions converge is more pronounced, it will be easier to locate the highest curvature point and extract all necessary information to differentiate between a *PA* and a *PF*. With regards to these necessary information, it was found that the smallest width w_c of a region lies on the row where all regions converge, as seen in Figure 3.44. This region is referred to as the innermost region. The maximum width w_e taken below the converging area of the innermost region, also differ for a *PA* and a *PF*. This is based on the amount of orientation fields flowing in the same direction at the edge of the fingerprint. For a *PA*, width w_e is more than half of the width of the *DP*. Conversely, for a *PF*, width w_c is fairly similar in length to width w_e . Therefore, the differences in these widths are unique for a *PA* and a *PF*.

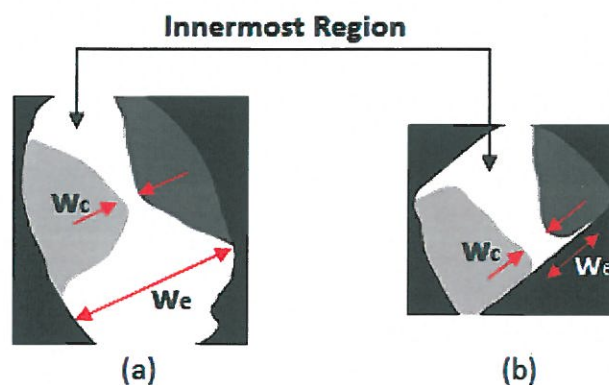


Figure 3.44: Width w_c is the smallest width on the innermost region and width w_e is the maximum width below the converging area on (a) a *PA* and (b) a *PF*

3.5 Recommendation

Based on the observations made in this chapter, the optimal number of regions is 3, since it produces the least amount of visible noise present on the *DP* and it outputs the highest *SP* detection accuracy irrespective of the value of *PT* and *ND* [8]. Table 3.3 shows a summary of the rotation recommendations for each flat fingerprint case, to produce consistent and unique patterns.

3.6 Conclusion

Small inter-class variability across classes is a major concern especially when flat fingerprints are not completely captured, which results in missing *SPs*. Achieving consistent unique *DPs* can reduce the inter-class variability issues across flat fingerprint classes. It was found that the number of regions and method of rotation, directly impacts the *DP* produced, in terms of its uniqueness and consistency.

To obtain a unique *DP* which best represents the classes, an investigation was undertaken to determine the optimal number of regions and optimal method of rotation for each flat fingerprint case. Based on this investigation, it was established that a 3-region *DP* for all classes produced the highest *SP* accuracy of 92.1%. Furthermore a 3-region *DP* has large regions, which allows easy detection and removal of visible noise.

From the investigation it was also found that each class produced inconsistent patterns under rotation. After conducting further analysis on the *DPs* under rotation, unique patterns appeared when the classes with the same number of *SPs* were rotated in a consistent manner. Classes with a single loop and single delta can be rotated such that a loop and delta are vertically aligned to make the *DPs* for each class unique. For single loop classes, unique patterns appear when the image is aligned such that the loop direction points downwards. It was found that unique patterns for single delta classes appeared when the images were rotated such that the orientation fields are horizontal below the delta. For fingerprints with no *SPs*, the orientation fields on the left should point downwards to highlight the highest curvature point on the fingerprint. The information around highest curvature point can be used to differentiate a *PA* from a *PF*.

Table 3.3: Recommendation for the method of rotation used to obtain unique *DPs* for each class of a given flat fingerprint case

Number of SPs	Two loops and/or two deltas	One loop and one delta			One loop	One delta	No SPs
Flat fingerprint type	Complete/incomplete	Complete			Incomplete	Incomplete	Complete/incomplete
Class	<i>W</i>	<i>RL</i>	<i>LL</i>	<i>TA</i>	<i>RL</i>	<i>LL</i>	<i>PA</i> <i>PF</i>
Method of rotation to obtain unique DP	No alignment required	Vertically align <i>SPs</i>			No alignment required	Loop direction must point downwards (90°)	Rotate so that orientation fields on the left of the fingerprint should point downwards
						Rotate so that orientation fields below the delta are horizontal	

Chapter 4

A Missing Singular Point Resistant Fingerprint Classification Technique

This chapter covers the implementation of the proposed exclusive fingerprint classification that is specifically designed to classify fingerprints with missing SPs. The remaining SPs found on the fingerprints are used to rotate the fingerprint to establish unique DPs. The algorithm is based on the rule-sets derived from these unique DPs for each class.

4.1 Introduction

Based on the literature study conducted in this dissertation, the most appropriate feature to classify fingerprints faced with issues of missing *SPs* is *DPs*. Since a *DP* is created by grouping orientation fields into regions that can be globally analysed, it provides more details with regards to the structural attributes of the classes. On that account, it has the potential to classify fingerprints with missing *SPs*. The remaining *SPs* found on these fingerprints can also be used to rotate the image appropriately, to produce unique patterns for each class. Even though these features show potential in solving the issues of compact scanners, previous exclusive fingerprint classification methods that are designed to solve flat fingerprint cases, have not undertaken any explicit study to determine whether these features can classify fingerprints with missing *SPs*. Hence, the purpose of this study is to determine if *DPs* combined with the remaining *SPs* are capable of overcoming this challenge in exclusive fingerprint classification. This is achieved by proposing a fingerprint classification algorithm that uses both *DPs* and the remaining *SPs* found on the fingerprint to counteract the issues of missing *SPs*. In this chapter, the details of the implementation of the proposed fingerprint classification algorithm is discussed.

The classification algorithm presents a rule for each flat fingerprint case. There are various flat fingerprints cases with different numbers of loops and deltas. Table 4.1 indicates which flat fingerprint cases were addressed by previous techniques and which are addressed by the proposed classification method. Although some cases have been addressed by previous techniques, not all were successful in terms of preserving rotation and overcoming inter-class variability issues [7, 10, 19, 44]. Therefore, new rules using *DPs* and *SPs* are created.

Table 4.1: A list of all the possible flat fingerprint cases addressed by previous literature and the proposed method

No. of loops	No. of delta	Class	Attempted by previous literature	Attempted by proposed classification technique
2	2	W	Yes	Yes
2	1	W	Yes	Yes
2	0	W	Yes	Yes
1	2	W	Yes	Yes
0	2	W	Yes	Yes
1	1	W	No	Yes (top and bottom loop)
		TA	Yes	Yes
		RL	Yes	Yes
		LL	Yes	Yes
1	0	W	No	Yes (top loop)
		TA	No	Yes
		RL	Yes	Yes
		LL	Yes	Yes
0	1	W	No	No
		TA	No	No
		RL	Yes	Yes
		LL	Yes	Yes
0	0	PA	Yes	Yes
		PF	Yes	Yes

The investigation in Chapter 3 established a consistent number of regions and a consistent method of rotation using *SPs* to rotate the fingerprint image in a specific way that can be repeated. The method of rotation also produces unique *DPs* for each flat fingerprint case. The rules are based on these unique patterns.

The following sections present the classification rules and their implementation. Section 4.2

introduces the set-up used for the fingerprint classification scheme. The details of the *DP* formation and *SP* detection, which are specifically used to implement this technique, are also presented in this section. The implementation of the classification rules, is covered in Section 4.3. Section 4.4 discusses the details of the testing design, the software used for the experimental testing, as well as the quantitative assessment. The conclusions are drawn in Section 4.5. The methodology aims to illustrate the classification technique, which will be used to prove that *DPs* and *SPs* are capable of classifying fingerprints that have missing *SPs* as a result of not being fully captured by electronic scanners.

4.2 Classification Set-up

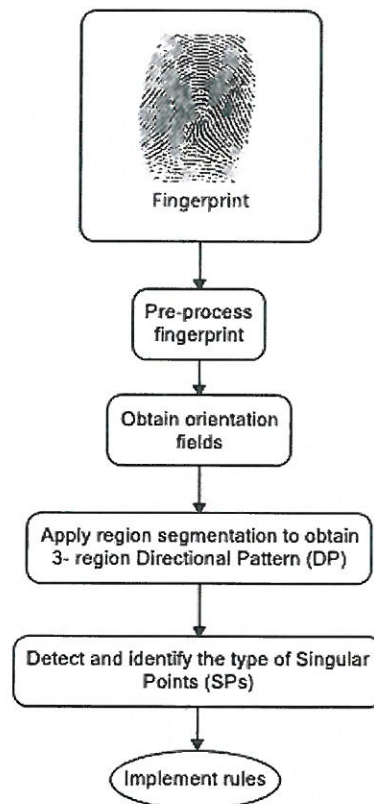


Figure 4.1: Brief overview of the complete classification process of the proposed algorithm

Figure 4.1 shows the flow diagram of the different stages of the proposed classification technique.

There are five major stages, namely:

1. Pre-processing
2. Orientation field estimation
3. Region segmentation
4. Detection and identification of the type of *SP*
5. Implementation of classification rules

For this implementation, a rule-based approach is employed. A rule-based approach is composed of numerous sets of conditions that satisfy the characteristics of a fingerprint class. Unlike machine learning techniques, a rule-based approach does not require a training *DB*.

4.2.1 Pre-processing and orientation field estimation

For the pre-processing of images and orientation field estimation, the same segmentation technique and orientation estimation algorithm covered in Chapter 3 is used.

4.2.2 Region segmentation

The evaluation in Chapter 3 proves that the optimal number of regions is 3. A 3-region *DP* produces the most distinct patterns. In addition, the *DP* with 3 regions was also found to have the highest *SP* detection accuracy, since it did not experience multiple intersecting points of 3 and 4 regions. Having a high *SP* accuracy is very important, considering that the number and type of *SP*s is used to determine which classification rule has to be performed on the input fingerprint. On that account, for this classification technique, the *DP* is partitioned into 3 regions.

To formulate a 3-region *DP*, the specific number of ranges (n), interval ($\Delta\phi$), range values ($range_i$) and the region number ($region_{num}$) for each range has to be calculated using Equations 3.10 to 3.12. Table 4.2 shows the final values that are used to develop the *DP*, by using the orientation matrix (O).

Table 4.2: Calculated values that produce a three region segmentation

n	$\Delta\phi$	i	$range_i$	$region_{num}$
3	60	1	0:60	1
		2	61:90	2
		3	91:180	3

An example of a 3-region *DP*, using the conditions listed in Table 4.2, is shown in Figure 4.2. Since the *DP* is partitioned into 3 regions, the size of each region is larger and there is a higher

possibility of having less visible noise present on the *DP*. When noise is found on the *DP*, it will appear as isolated regions that lie within a larger region. These isolated regions are not attached to the intersecting point like other regions in the *DP*. Removing the isolated regions decreases the risk of detecting false *SP*s. The isolated regions can be easily identified and removed. To remove them, they are simply assigned to the region number of the surrounding region.

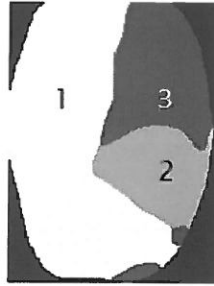


Figure 4.2: An example of a *DP* with three different regions each with its own orientation range [7]

4.2.3 *SP* detection

Based on the experiment conducted in Chapter 3, the *ND* value which produces a high *SP* detection accuracy for 3 regions is 24. This *ND* value will be used to detect the *SP*s. Figure 4.3 depicts an example of a 5×5 matrix (24 *ND*) with intersecting point at $region_{num}(r, c)$. The intersecting point of all 3 regions is the *SP*. By searching for 3 different pixel values within the *ND*, the *SP* can be detected.

Once the *SP* is located, the type of *SP* is then identified. By detecting the direction of the region number from 1 to 3, within the 5×5 matrix around the *SP*s, the type of *SP* can be determined. When regions move from 1 to 3 in an anti-clockwise direction, it is a loop. Whereas, when regions move from 1 to 3 in a clockwise direction, it is a delta. The number of loops and deltas found on the fingerprint is associated with a specific rule. Each rule classifies fingerprints into possible classes that can occur for a given number of loops and deltas.

4.3 Implementation of classification rules

Table 4.3 contains the details of the unique patterns for each flat fingerprint case with its associated rule used for the classification scheme. Amongst the flat fingerprint cases listed in Table

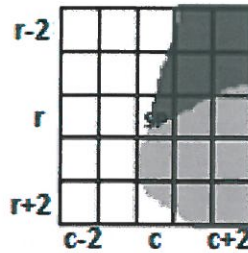


Figure 4.3: Intersection of 3 regions that represents a *SP* using a 24 *ND* [7]

4.3, *W* fingerprints with two loops or two deltas, is the only class that has a unique number of *SP*s. Therefore, when two loops or two deltas are detected, the fingerprint can be immediately classified as a *W*. However other flat fingerprint cases require additional rules when a specific number of loops and deltas are identified. These classification rules are derived from the unique *DP*s obtained by rotating a fingerprint of a given number of loops and deltas in a specific way. Figure 4.4 shows the overview of the entire classification algorithm and sub-algorithms represented by 1, 2, 3 and 4. Each sub-algorithm represents the rules for the flat fingerprint cases of a given number of loops and deltas. The rules are developed to classify an *RL*, *LL*, *TA*, *PA*, *PF*, *W* and an unclassifiable class (*U*). A fingerprint containing *l* loops and *d* deltas is class *U* if it does not meet the conditions within a rule set that is designed for *d* deltas and *l* loops.

4.3.1 Flat fingerprint case containing a loop and a delta

A detailed decision tree depicting the overview of the rules for flat fingerprint cases with a single loop and single delta is shown in Figure 4.5. The rule can identify a *W*, *RL*, *LL* and *TA* with a single loop and delta. There are two types of loops, namely: top and bottom. Amongst these classes, *W*s are the only class that have a bottom loop. The difference between a bottom loop and a top loop, is the direction of the orientation fields about the loop. The direction of the orientation fields move downwards as it approaches the bottom loop, as shown in Figure 4.7 (a). Since it is a unique characteristic of a *W*, when a bottom loop is detected the fingerprint can be immediately classified as a *W*. A *W* with a single top loop and delta is unlikely to occur. The structural layout of a *W* makes it difficult to not capture the bottom loop, since it lies between the top loop and either one of the deltas, as shown in Figure 4.6. In addition to this, the *DB* did not contain such cases were a *W* had a top loop and a delta. Therefore, only a rule for a *W* with a bottom loop and a single delta was developed.

Table 4.3: Overview of each flat fingerprint case rule that is created based on the unique property established through rotation

Number of SPs	Two Loops or Two Deltas	One Loop and One Delta			One Loop	One Delta	No SP
Flat fingerprint type	Complete/incomplete	Complete			Incomplete	Incomplete	Complete/incomplete
Class	W	RL	LL	TA	W	RL	LL
Method of rotation to obtain unique DP	No alignment required	Vertically align SPs			No alignment required	Loop direction must point downwards (90°)	Rotate so that orientation fields at point of interest should point downwards
Unique Property	Unique number of SPs	CRs has unique region numbers			Region 2 lies above loop	Region numbers below region 2 differ	Width at the convergence point versus width at the edge of the fingerprint
Overview of Rule	Based on the number of SPs	Based on the CRs region numbers			Based on the position of region 2 relative to the loop	Based on the side of the loop a certain region number lies	Based on the width of the innermost region on the DP

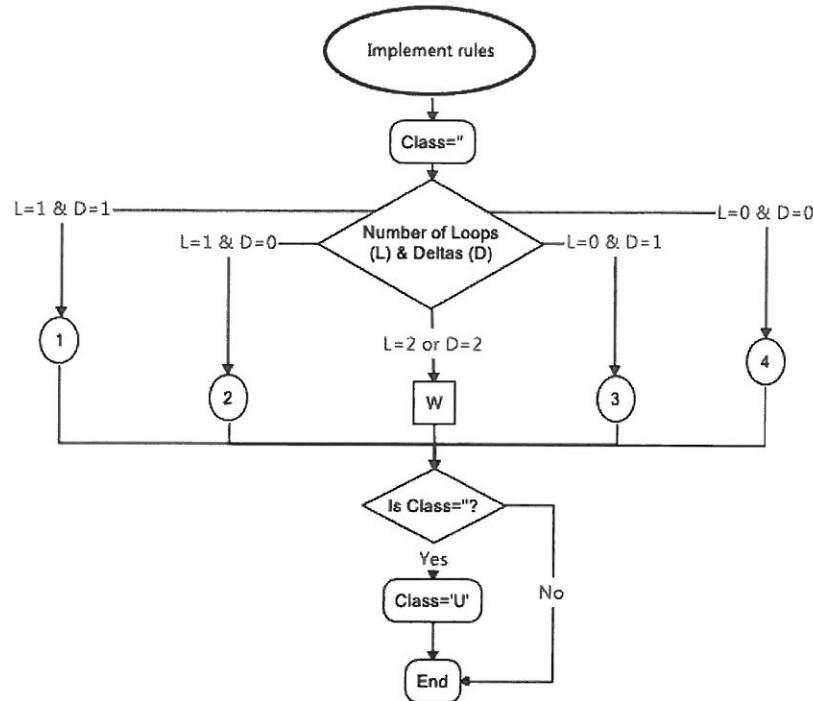


Figure 4.4: Decision tree of the overview of the proposed fingerprint classification algorithm, with output *W* and sub-algorithms indicated by 1, 2, 3 and 4

4.3.1.1 Detecting a bottom loop

To detect the bottom loop, the rule observes if a specific region lies above the loop. Each region represents orientation fields falling within a specific range. Since the direction of the orientation fields move downwards as it approaches the bottom loop, orientation fields fall within a range of approximately 61° to 90° above the bottom loop. This range represents region 2. Therefore, region 2 lies above the loop for a bottom loop. Whereas for a top loop, region 2 lies below the loop. It was also found that region 2 is largest region on a *W* and would require the fingerprint to be rotated almost 50° from its upright position to change the region number to 1 or 3. Such a rotation is unlikely to occur, since the fingerprints in the testing *DBs* are limited to a maximum rotation of 35° . Hence, there is no alignment of *SPs* applied on the fingerprint. The rule to identify a bottom loop from a top loop, simply detects if region 2 lies above the loop.

The width of region 2 is also taken into account when detecting a bottom loop. This is because for both complete and incomplete *Ws*, region 2 covers almost the entire width of the fingerprint

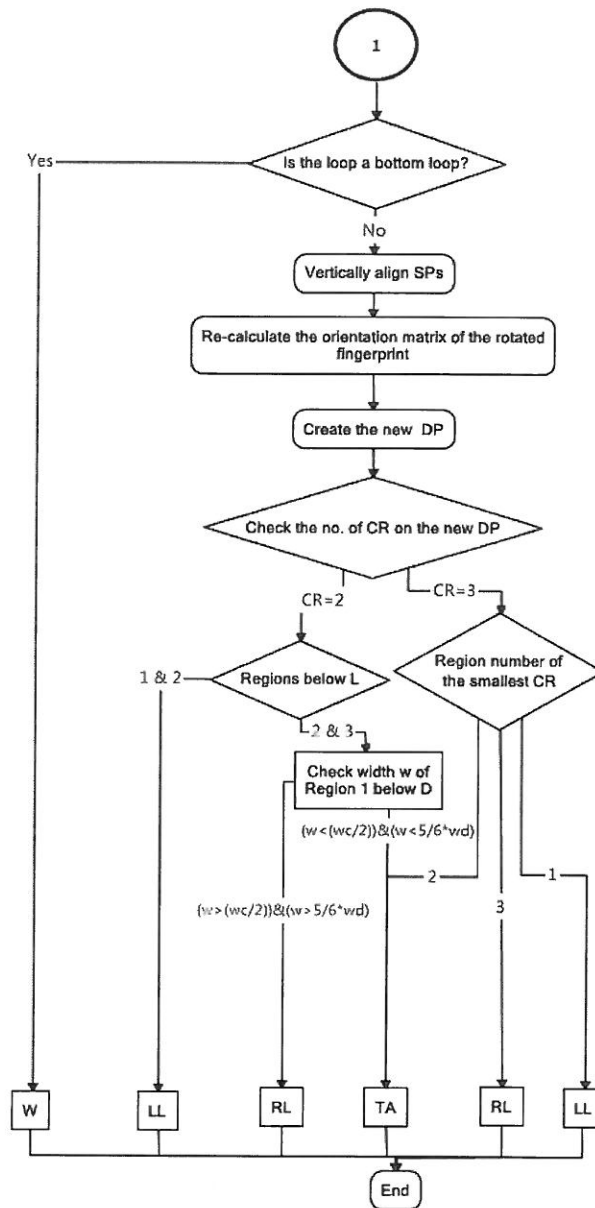


Figure 4.5: Decision tree of fingerprint classification specific to a one loop one delta rule with class outputs, *RL*, *LL*, *TA* and *W*

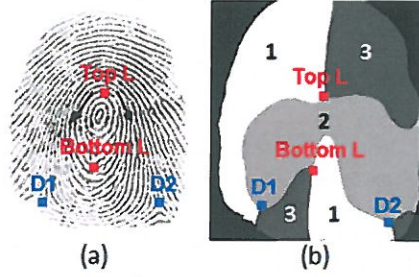


Figure 4.6: A complete W with two loops and two deltas, indicating a top loop and a bottom loop on (a) a segmented fingerprint and (b) a DP

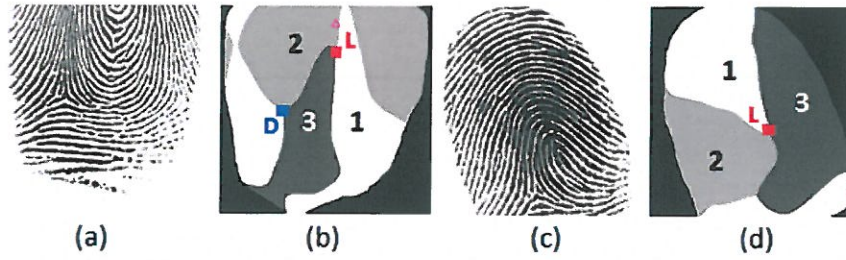


Figure 4.7: A DP illustrating a single bottom loop L in class (a) W and a single top loop L in class (b) RL

due to the symmetrical structure between the loops and deltas, as seen in Figure 4.7. Therefore, for a bottom loop to exist, the width of region 2 must be considered in addition to region 2 lying above the loop. To accommodate noise near the boundary of the fingerprint, the width of region 2 has to be greater than $3/4$ of the fingerprint width.

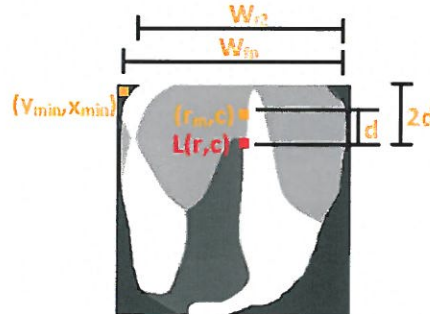


Figure 4.8: Symbols depicted on a single loop and single delta W , that are used to perform the detection of the type of loop

Algorithm 1 is used to determine a W with a bottom loop and a delta. Variables w_{fp} , w_{r2} , r_m , $Region$, y_{min} , r , and c are used in algorithm 1 and are shown as symbols in Figure 4.8. The

region number at the midpoint between the top edge of the fingerprint and loop $L(r, c)$. The width of region 2 is used to identify a top loop from a bottom loop.

Algorithm 1 Classify a W with a bottom loop and a delta

Input: $w_{fp}, w_{r2}, r_m, Region, y_{min}, r, c$

Output: $class$

```

1:  $class = ''$ 
2:  $r_m = (r + y_{min})/2$ 
3: if ( $Region(c, r_m) == 2$ ) & ( $3/4 * w_{fp} < w_{r2}$ ) then
4:    $class = 'W'$ 
5: end if

```

When the class output is a null, the general rule that classifies an RL , LL and TA with a single loop and a single delta is performed. For these classes the fingerprint SP s are vertically aligned before classification can occur.

4.3.1.2 Vertically aligning SP s for a fingerprint with a single loop and single delta

The SP s are vertically aligned to achieve unique patterns in fingerprints containing a single loop and single delta. The first step is to obtain the angle β between the loop and delta. The segmented fingerprint image is then rotated by $\beta - 180^\circ$. This vertically aligns SP s as shown in the Figure 4.9. The new β value is 180° .

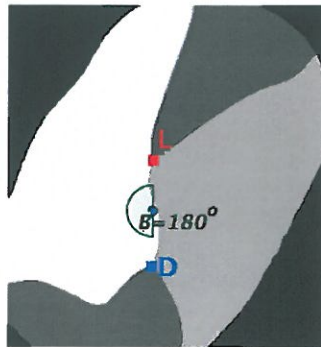


Figure 4.9: DP produced from a LL fingerprint where the loop (L) and delta (D) are vertically aligned [7]

After the SP s are vertically aligned, a new DP is formed using the rotated fingerprint. The alignment produces unique patterns for an LL , RL or TA . Two layouts are produced for each class. These are two CR and three CR layouts. Rules for each type of layout were developed.

4.3.1.3 Rule for DPs with two CRs

In this case, two *CRs* exist between the *SPs*. Figure 4.10 consist of the different *DPs* produced from an *LL*, *RL* and *TA* fingerprint with a loop *L* that is vertically aligned with the delta *D*. To differentiate between these classes, the region numbers are analysed [7]. When the *CRs* are region number 1 and 2, the fingerprint is immediately classified as a *LL* and no additional rules are needed. For an *RL*, the most common arrangement of the *CRs* is that region 2 lies on the right of the *DP* and region 3 lies on the left of the *DP*. Whereas for a *TA*, the most common arrangement of the *CRs* is that region 2 lies on the left of the *DP* and region 3 lies on the right. However in some cases, for an *RL* region 2 lies on the left of the *DP* and region 3 lies on the right of the *DP*. This is the same *CR* arrangement of a *TA*. Therefore a rule has been included for this case. There is a possibility that a *TA* may appear to be very similar to an *LL* but the rule for this case has not been included because cases like this have not been found in the *DB*. As a result, a rule is only created for a *TA* and *RL* with a 2-*CR* layout.

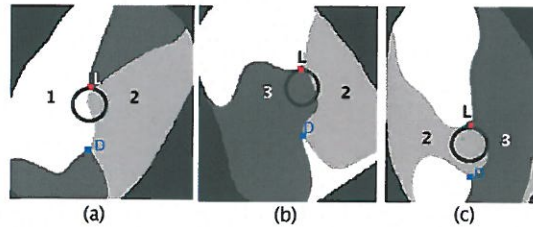


Figure 4.10: Two *CR* layouts formed from classes (a) *LL*, (b) *RL* and (c) *TA* [7]

To differentiate an *RL* from a *TA*, the region numbers below the delta is observed since it is unique for each of these classes. These region numbers found below the delta, represent the direction of the ridges as they enter and exit the aligned fingerprint. Since the orientation fields move left to right below the delta for a rotated *RL*, the region number is 1. However for a *TA* the region numbers are 1 and 3, since the orientation fields below the delta move in opposite directions. The *DP* is then realigned to the fingerprints original rotation, so that the width can be obtained along side the x axis. Thereafter, the maximum width of region 1 and 3 that lies below the midpoint (*M*) are obtained. Variable *M* is the midpoint between *D* and the bottom edge of the *DP*. The symbols for the maximum widths that lie below *M* are, w_f , w_1 and w_3 which represent the width of the fingerprint, region 1 and region 3, respectively [7]. Variable w_1

is the width that is taken from the first occurrence of region 1 to the last occurrence of region 1, regardless whether region 1 is not connected. These symbols are depicted in Figure 4.11.

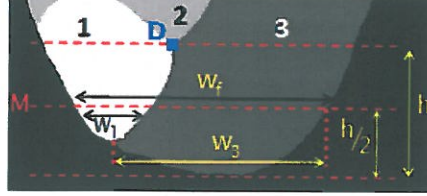


Figure 4.11: The maximum width w_f , w_1 and w_3 for a TA [7]

Algorithm 2 provides the procedure used to classify a DP that produces a two CRs layout. Variables w_f , w_1 , w_3 and CR are adopted to perform algorithm 2.

Algorithm 2 Classify a 2-CR layout [7]

Input: CR, w_f, w_1, w_3

Output: *class*

```

1: if ( $CR=1$ ) then
2:   class = 'LL'
3: else if ( $CR=3$ ) then
4:   // Check the area below the delta is mostly composed of region 1
5:   if ( $w_1 > (1/2 * w_f) \& (w_1 > w_3)$ ) then
6:     class = 'RL'
7:   else
8:     class = 'TA'
9:   end if
10: end if

```

4.3.1.4 Rules for DPs with three CRs

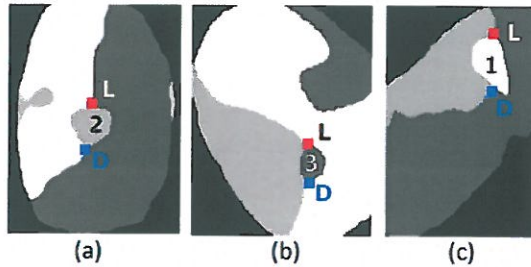


Figure 4.12: 3-CR layout produced by classes, (a) TA, (b) RL and (c) LL [7]

For a layout with a 3-CR, the region number of the smallest CR (CR_{sr}) is unique for an LL, RL and TA [7]. Figure 4.12 shows an LL, RL or TA, with a CR_{sr} of 1, 2 and 3, respectively.

On that account, if a three CR s is detected, the region number of CR_{sr} is used for classification. Algorithm 3 is used to classify an LL , RL or TA that has three CR s.

Algorithm 3 Classify a 3- CR layout [7]

Input: CR_{sr}

Output: $class$

```

1: // Check the region number of  $CR$  to determine the class
2: if ( $CR_{sr}=1$ ) then
3:    $class='LL'$ 
4: else if ( $CR_{sr}=2$ ) then
5:    $class='TA'$ 
6: else ( $CR_{sr}=3$ )
7:    $class='RL'$ 
8: end if

```

4.3.2 Flat fingerprint case with a single loop

The overview of the rule for a fingerprint with a single loop is shown in Figure 4.13. A fingerprint captured with a single loop could be an RL , LL , TA without a delta, or a W without a loop and both deltas. The DP is initially analysed to determine if the loop is a top loop or bottom loop. The rule for detecting a bottom loop was covered in Section 4.3.1.1. If a bottom loop occurs, it is a W . If not then the fingerprint is rotated. A new DP is formed from the rotated fingerprint. The classification is applied to this new DP .

4.3.2.1 Rotation of fingerprints with a single loop

The loop direction for a segmented fingerprint with a single loop must point downwards to obtain a unique DP .

To calculate the loop direction, the balance arm flow method of Guo *et al.* is employed [7, 44]. Using the average orientation fields around the loop area the method tracks the points that lie along the balance arm line. These point can be used has an indication of the direction of the loop. Guo *et al.* [44] stated that the innermost ridge line of a loop is referred to as the balance arm line. The direction of the balance arm line of the segmented fingerprint is illustrated by the arrow in Figure 4.14 (a). Using eleven points to calculate loop direction is sufficient. This value was established during the experimental stage. After calculating the loop direction, it is rotated to be 90° . Figure 4.14 (b) shows the loop direction aligned to 90° .

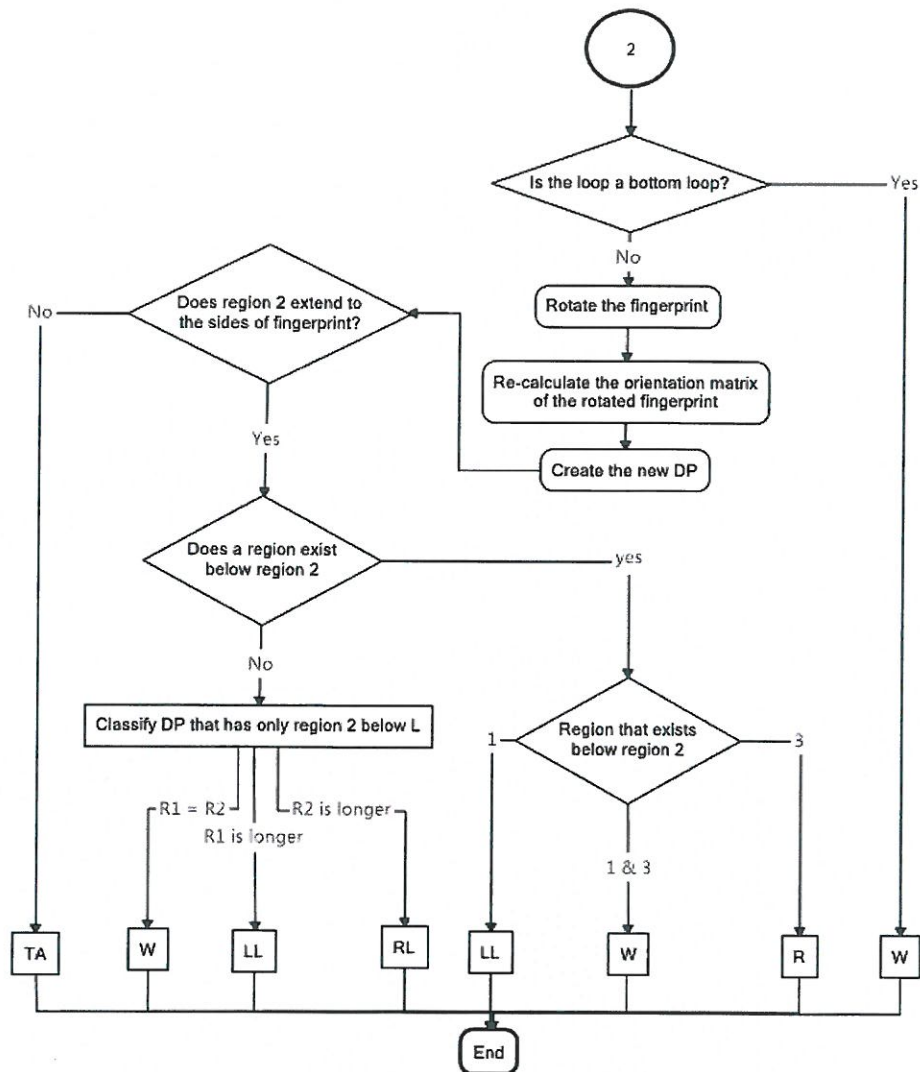


Figure 4.13: Decision tree of fingerprint classification algorithm for a single loop with class outputs, W, RL, LL, and TA

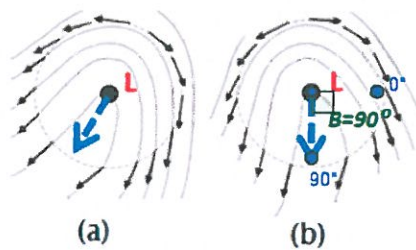


Figure 4.14: Loop direction pointing (a) 135° and (b) 90° [7]

4.3.2.2 Rules to classify a DP with a single loop

For a rotated *RL*, *LL* and *W* fingerprint, the flow of the orientation fields below the loop all move downwards. These ridges represent region 2. Region 2 forms the largest region which extends to either side of the *DP*. However, *TA* is the only class where its region 2 does not extend to the outer limit of fingerprint as shown in Figure 4.15. Therefore, a rule is created to detect whether region 2 touches to side limits of the fingerprint. If region 2 does not touch the edge of the fingerprint, it is immediately classified as a *TA* without the need for additional rules. For *RL*, *LL*, and *W*, the *DP* produces unique region numbers below region 2 as shown in Figure 4.15. This is owing to the fact that the ridge flow of each class is unique. To classify a *DP* with a single loop, the rule identifies region numbers below region 2 of an *RL*, *LL* or a *W* with a top loop.

To observe the region numbers below region 2, the pixels along lines CL_1 and CL_2 are observed. These lines are used to determine the fingerprint class as shown in Figure 4.15.

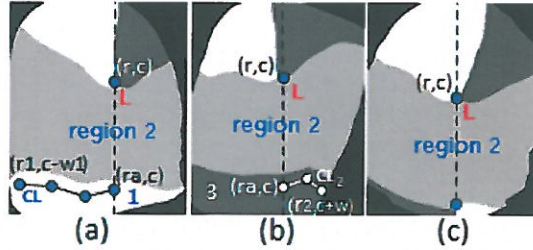


Figure 4.15: *DP* containing a single loop (a) *LL*, (b) *RL* and (c) *W* [7]

The lines CL_1 and CL_2 extend from the loop column c and row r_a to the left and right edge of the *DP*. The lines follow the curvature of the region boundary, depicted in Figure 4.15 as w_1 and w pixels. The row r_a lies one pixel below the division of region 2.

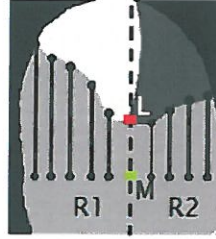
When the pixels that represent region 1 are detected on line CL_1 at $Region(r_1, c - i_1)$ for $i_1 = 0 \dots w_1$, the *DP* is classified as a *LL* [7]. Whereas, when the pixels that represent region 3 are detected on line CL_2 at $Region(r_2, c + i)$ for $i = 0 \dots w$, the class is a *RL*. Unlike *RLs* and *LLs*, *Ws* are the only class that has two regions (1 and 3) forming below region 2 as a slight appearance of its bottom loop. It is important to note that region 3 will only lie on the left of column c and on the right of column c lies region 1 [7]. Algorithm 4 classifies a *LL*, *RL* and *W* with a single loop, using variables r_1 , r_2 , c , w_1 and w .

Algorithm 4 Classify a *DP* with a single loop fingerprint [7]**Input:** r_1, r_2, c, w_1, w **Output:** *class*

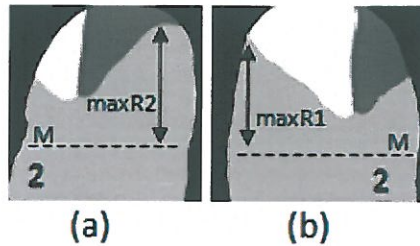
```

1: if for all  $i_1 = 0 \dots w_1, \text{Region}(r_1, c - i_1) = 1$  then
2:   class = 'LL'
3: else if for all  $i = 0 \dots w, \text{Region}(r_2, c + i) = 3$  then
4:   class = 'RL'
5: else if (for all  $i_1 = 0 \dots w_1, \text{Region}(r_1, c - i_1) = 3$ ) or
6:   (for all  $i = 0 \dots w, \text{Region}(r_2, c + i) = 1$ ) then
7:   if (for all  $i_1 = 0 \dots w_1, \text{Region}(r_1, c - i_1) \neq 1$ ) &
8:   (for all  $i = 0 \dots w, \text{Region}(r_2, c + i) \neq 3$ ) then
9:     class = 'W'
10:  end if
11: end if

```

**Figure 4.16:** A single loop *DP* that has no regions below region 2 [7]

For cases where there is not enough information below the loop, the regions below region 2 may not appear. This requires an additional rule. It is found that the length of region 2 on either side of the loop differ for each class. Therefore, region 2 is divided at loop *L* into subregions *R1* and *R2* as shown in Figure 4.16. The length *maxR1* and *maxR2* lie within *R1* and *R2*, respectively. These lengths are used to determine the class type. The length *maxR1* and *maxR2* is taken from the upper boundary of region 2 to the midpoint (*M*) [7]. Variable *M* lies between the loop *L* and the bottom edge of the fingerprint image [7].

**Figure 4.17:** *DP* of class (a) *RL* with *maxR2* longer in length and (b) *LL* with *maxR1* longer in length

Owing to the symmetrical structure of Ws , $maxR1$ and $maxR2$ are similar in length. However, the value for $maxR1$ and $maxR2$ differ for a RL and a LL . For RLs , $maxR2$ is longer than $maxR1$ as seen in Figure 4.17 (a). Whereas for a LL , $maxR1$ is longer than $maxR2$ as depicted in Figure 4.17 (b). The reason for this is that more ridges flow in the direction of the loop. Algorithm 5 is used to classify a single loop DP with no region below region 2.

Algorithm 5 Classify a single loop with only region 2 below L [7]

Input: $R1, R2$

Output: $class$

```

1: // Obtain the line with the largest length of line
2: // found in  $R1$  and  $R2$  and store
3: // lengths into  $maxR1$  and  $maxR2$ , respectively.
4: if  $abs(maxR1 - maxR2 < 10)$  then
5:    $class = 'W'$ 
6: else if  $maxR1 > maxR2$  then
7:    $class = 'LL'$ 
8: else if  $maxR2 > maxR1$  then
9:    $class = 'RL'$ 
10: end if

```

4.3.3 Rule for flat fingerprint cases with a single delta

This rule classifies an RL or an LL with a single delta. An overview of the single delta rule is depicted in the decision tree shown in Figure 4.18.

4.3.3.1 Rotation of fingerprints with a single delta

In order for the patterns of an LL and RL containing one delta to be unique, the flow of orientation fields below the delta of the segmented fingerprint should be horizontal. An example of a flat fingerprint with a single delta D that has been rotated such that the average orientation fields below D are horizontal is shown in Figure 4.19 (b). The flow of orientation fields are indicated by the arrow within the circle.

To perform rotation, the average orientation fields below delta $D(r, c)$ must be calculated. Regions in which to calculate orientation fields are identified using points of interest, namely: the location of the delta $D(r, c)$, points A_L at point (r, c_1) , A_R at point (r, c_2) and A_B at point (r_1, c) that lie on the boundary of the fingerprint, and the midpoints A_{Lm} at point (r, c_{1mid}) , A_{Rm} at point (r, c_{2mid}) and A_{Bm} at point (r_{1mid}, c) as shown in Figure 4.20.

The midpoint A_{Lm} lies between A_L and delta $D(r, c)$. The midpoint A_{Rm} lies between A_R and delta $D(r, c)$. Lastly, the midpoint A_{Bm} lies between A_B and delta $D(r, c)$. Equations 4.1 to 4.4

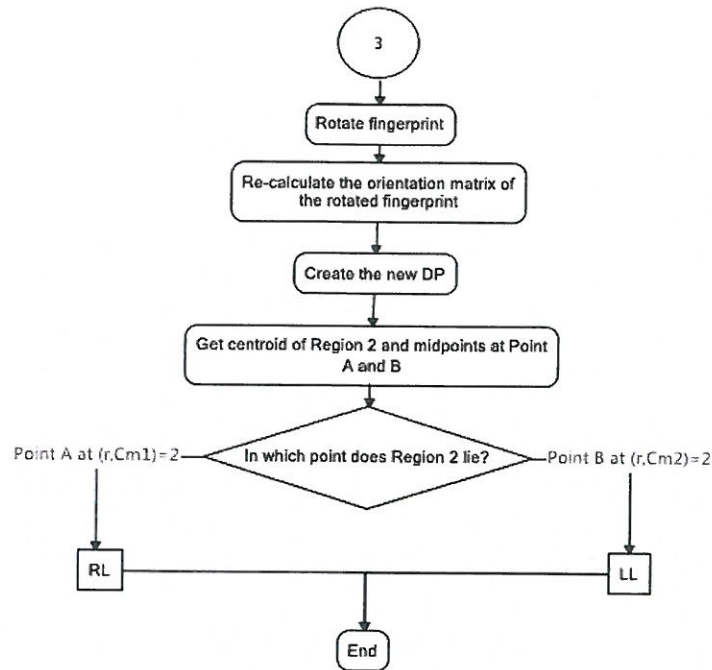


Figure 4.18: Decision tree of fingerprint classification algorithm for a single delta with class outputs *RL* and *LL*



Figure 4.19: Fingerprint with single delta *D* where the orientation fields below *D* flows (a) appropriately 35° and (b) horizontal

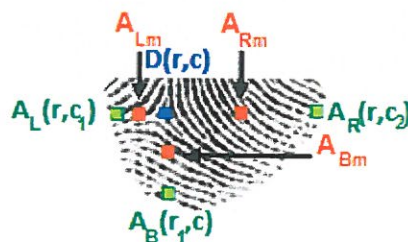


Figure 4.20: Points of interest on a fingerprint with single delta *D*

are used to calculate the average orientation fields on the right and left of the delta.

$$Avg_{right} = \frac{\sum_{i=c_{2mid}}^{i=c_{2mid}+10} [O(r_{1mid}, i)]}{10} \quad (4.1)$$

$$Avg_{left} = \frac{\sum_{i=c_{1mid}}^{i=c_{1mid}+10} [O(r_{1mid}, i)]}{10} \quad (4.2)$$

Since matrix O consists of orientation fields in radians, the final angle is converted to degrees. ρ_R and ρ_L are the converted versions of Avg_{right} and Avg_{left} and are calculated using Equation 4.3 and Equation 4.4. These are used to determine how much the image needs to be rotated in order to make the orientation flow below the delta horizontal. This is achieved using Algorithm 6.

$$\rho_R = Avg_{right} * (180/\pi) \quad (4.3)$$

$$\rho_L = Avg_{left} * (180/\pi) \quad (4.4)$$

4.3.3.2 Rule for DPs with a single delta

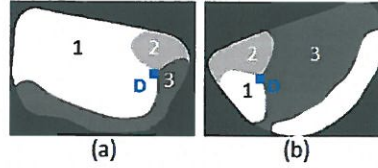


Figure 4.21: A single delta DP illustrating the position of region 2 for, (a) a LL and (b) a RL

After the fingerprint is rotated, the orientation matrix is re-calculated and used to form the new DP . The most obvious pattern seen in Figure 4.21, is that the region number of the largest region is unique to a RL or LL . However, using the largest region for classification may result in the rule being too dependent on the amount of information captured. Therefore, a more robust rule that is independent on the size of region had to be developed. To determine whether the fingerprint with a single delta is a RL and LL , the rule identifies the side of the fingerprint in which the centroid of region 2 lies. When the centroid of region 2 lies on the right of the fingerprint, it is a LL . Whereas, if the centroid of region 2 lies on the left of the fingerprint it is a RL . This is illustrated in Figure 4.21.

Algorithm 6 Rotation of a fingerprint with a single delta**Input:** ρ_R, ρ_L **Output:** ρ_{rotate}

```

1:  $\rho_{rotate} = 0$ 
2: // Ensure that the quadrant of the average orientation
3: // field lies between 0 and 90.
4: if  $(\rho_R < 90) \& (\rho_L > 90)$  then
5:   Shift to the correct quadrant
6:    $\rho_L = 180 - \rho_L$ 
7:   if  $\rho_R < \rho_L$  then
8:      $\rho_{rotate} = (\rho_L - \rho_R)/2$ 
9:   else if  $\rho_L < \rho_R$  then
10:     $\rho_{rotate} = -[(\rho_R - \rho_L)/2]$ 
11:   end if
12: else if  $(\rho_R < 90) \& (\rho_L < 90)$  then
13:   // Since both  $\rho_R$  and  $\rho_L$  lie in the correct quadrant,
14:   // use the lower average field value as the angle of rotation  $\rho_{rotate}$ .
15:   if  $\rho_R < \rho_L$  then
16:      $\rho_R = \rho_L$ 
17:   end if
18:    $\rho_{rotate} = \rho_R$ 
19: else if  $(\rho_R > 90) \& (\rho_L > 90)$  then
20:   if  $\rho_R > \rho_L$  then
21:     // Shift to the correct quadrant
22:      $\rho_R = 180 - \rho_R$ 
23:   end if
24:   if  $\rho_R < \rho_L$  then
25:     // Shift to the correct quadrant
26:      $\rho_L = 180 - \rho_L$ 
27:      $\rho_{rotate} = \rho_R - \rho_L$ 
28:   end if
29:    $\rho_{rotate} = -[(\rho_R + \rho_L)/2]$ 
30: end if

```

This procedure is described in Algorithm 7, where the variables w_{fp} , c_c , c_{m1} , c_{m2} , $Region$ and x_{min} and r are used and are depicted in Figure 4.22 as symbols. The initial step of the rule is to locate the centroid (r, c_c) of the region 2. Points A and B represents the left and right side of the fingerprint. Points A and B are the midpoints from column c_c to the left and right edge of the fingerprint, located at $Region(r, c_{m1})$ and $Region(r, c_{m2})$, respectively. The width of the whole fingerprint is given by w_{fp} and minimum column value of the fingerprint mask is x_{min} .

4.3.4 Classifying DPs with no SPs

This rule is designed to classify a PA from a PF . An overview of the rule is shown in the flow diagram in Figure 4.23. The rule is more robust than the previous techniques since it doesn't

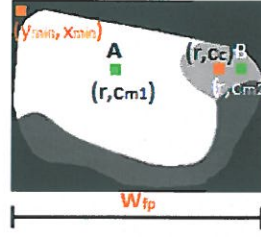


Figure 4.22: *DP* containing the symbols used to perform the classification rule of a single delta fingerprint

Algorithm 7 Classify a *DP* with a single delta

Input: w_{fp} , c_c , c_{m1} , c_{m2} , *Region*, x_{min} , r

Output: *class*

- 1: $c_{m1} = (c_c/2 + x_{min})$
 - 2: $c_{m2} = (w_{fp} - c_c)/2 + (x_{min})$
 - 3: **if** (*Region*(c_{m1} , r) == 2) **then**
 - 4: *class* = 'RL'
 - 5: **else if** (*Region*(c_{m2} , r) == 2) **then**
 - 6: *class* = 'LL'
 - 7: **end if**
-

depend on the size of the fingerprint but rather on the unique properties extracted from the actual patterns produced.

4.3.4.1 Alignment of fingerprints containing no SPs

Based on the findings in Chapter 3, the *DP* of a *PA* and a *PF* can be set apart by analysing the width of the innermost region at the point of convergence and the width of this region at the edge of the fingerprint. Therefore, the point where all region converge must be made clear as possible. This is achieved by rotating the fingerprint as shown in Figure 4.24. Since there are no *SPs* for a *PF* and a *PA*, a reference point for rotation needs to be selected. The centroid and the size of the fingerprint are used to determine the reference point.

The centroid of the fingerprint is located at column x_{ct} and row y_{ct} as shown in Figure 4.25. The height h_{fp} and width w_{fp} of the fingerprint are used to assist in obtaining the orientation fields around the regions of interest, *A* and *B*. Point *A* lies at position $(y_2, x_{ct} + x)$ and *B* lies at position $(y_2, x_{ct} - x)$. Equations 4.5 to 4.7 are formulae used to obtain the location of *A* and *B*:-

$$x = w_{fp}/4 \quad (4.5)$$

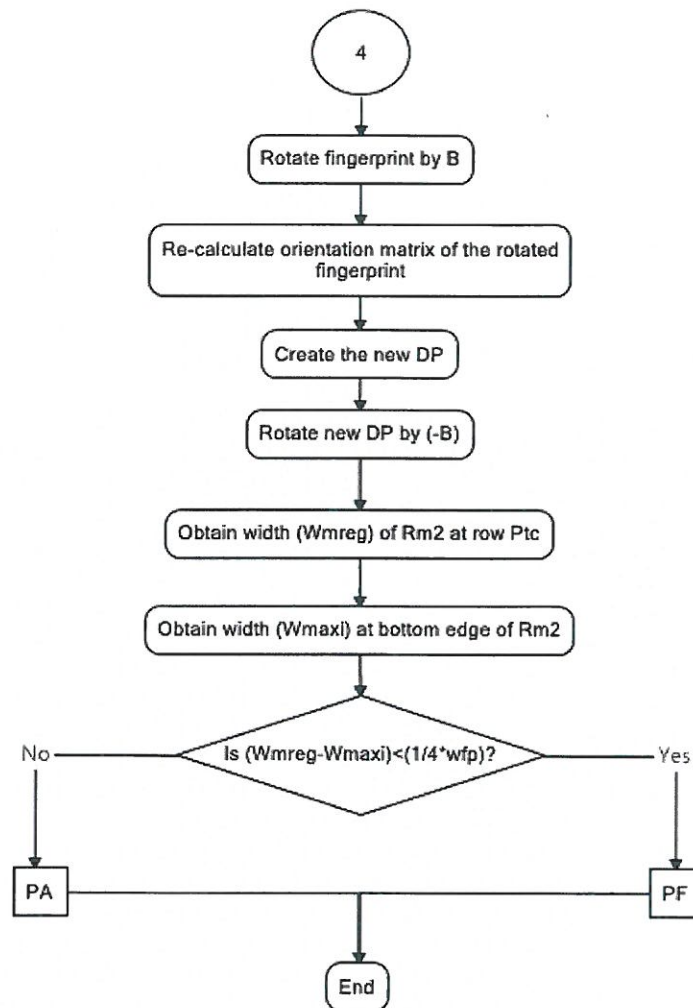


Figure 4.23: Overview of the rule used to classify a *PA* and a *PF*



Figure 4.24: Rotated *PF* where the orientation flow near point A is 90°

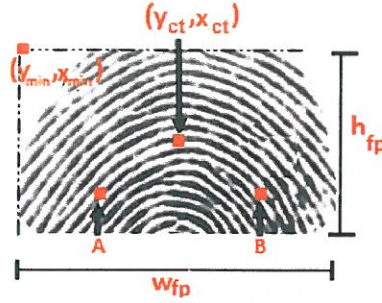


Figure 4.25: Points of interest on a PF fingerprint

$$y = h_{fp}/4 \quad (4.6)$$

$$y_2 = y_{ct} + y \quad (4.7)$$

The average orientation field is taken around the reference points, A and B. The equations are as follows:-

$$Avg_{right} = \frac{\sum_{i=y_2}^{i=y_2+10} [O(i, x_{ct} - x)]}{10} \quad (4.8)$$

$$Avg_{left} = \frac{\sum_{i=y_2}^{i=y_2+10} [O(i, x_{ct} + x)]}{10} \quad (4.9)$$

Since the values in the orientation matrix O are in radians, Avg_{right} and Avg_{left} are converted to degrees and stored in σ_R and σ_L using Equation 4.10 and Equation 4.11, respectively.

$$\sigma_R = Avg_{right} * (180/\pi) \quad (4.10)$$

$$\sigma_L = Avg_{left} * (180/\pi) \quad (4.11)$$

Figure 4.24 shows a rotated PF where the orientation fields near point A moves downwards (approximately 90°). To achieve this, a smaller angle between σ_R and σ_L is first selected to calculate the angle of rotation. The angle of rotation is represented by α . Algorithm 8 is used to obtain the angle of rotation α that will ensure that the orientation fields around point A is approximately 90° .

Algorithm 8 Rotation for a fingerprint with no SPs**Input:** σ_R, σ_L **Output:** α

```

1:  $\alpha=0$ 
2: //  $90^\circ$  points downwards on the orientation axis used.
3: // Shifting the image such that the fields around the delta
4: // are approx.  $90^\circ$ , diagonally aligns the fingerprint.
5: if  $\sigma_R < 90^\circ$  or  $\sigma_L < 90^\circ$  then
6:   // Use the smallest average value to achieve a larger angle of rotation.
7:   if  $\sigma_R < 90^\circ$  then
8:      $\alpha=90^\circ-\sigma_R$ 
9:   end if
10:  if  $\sigma_L < 90^\circ$  then
11:     $\alpha=90^\circ-\sigma_L$ 
12:  end if
13:  // Account for cases where the average orientation fields lie just above  $90^\circ$ .
14: else if  $\sigma_R > 90^\circ$  &  $\sigma_R < 100^\circ$  or  $\sigma_L > 90^\circ$  &  $\sigma_L < 100^\circ$  then
15:   if  $\sigma_R > 90^\circ$  &  $\sigma_R < 100^\circ$  then
16:      $\alpha=100^\circ-\sigma_R$ 
17:   end if
18:   if  $\sigma_L > 90^\circ$  &  $\sigma_L < 100^\circ$  then
19:      $\alpha=100^\circ-\sigma_L$ 
20:   end if
21: end if

```

4.3.4.2 Rule for a DP with no SPs

After rotation is applied to the fingerprint, the orientation matrix and the *DP* is re-calculated from the rotated fingerprint. The number of regions used to create the *DP* is increased for this rule. This ensures that the area where all regions converge is made as clear as possible. Using 3 to 7 regions will not be sufficient as it is still too few. Conversely, using a larger number of regions like 12 to 15 may experience visible noise. Therefore, the number of regions n used to create the *DP* is set to 10. An example of a 10-region *DP* that represents the rotated fingerprint for a *PA* and *PF* is shown in Figure 4.26. The width of the innermost region at the converging point compared to its width at the edge of the fingerprint differ for a *PA* and a *PF* class. For *PA*s, the regions narrows as it reaches the highest point of curvature and expands as it approaches to the edge of the fingerprint. This indicates that ridge flow from one side of the fingerprint to the other is continuous. Conversely, for a *PF* the regions generally get smaller in width as it reaches the edge of the fingerprint, since each region is approaching the point of intersection (loop), but the fingerprint has been cut-off before this point. These properties are used to classify the *PA* and *PF* class. The region R_m with the smallest width is shown in Figure 4.27. The rule detects

region R_m and then this region is used to identify if the fingerprint DP possesses characteristics of either a PA or a PF .

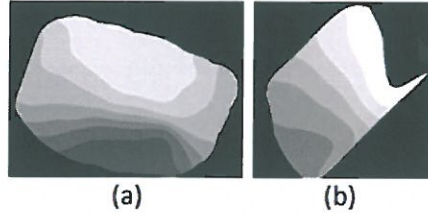


Figure 4.26: A 10-region DP that represents the rotated fingerprint for (a) a PA and (b) a PF

Figure 4.27 depicts a PF with minimum row y_{min} and minimum column x_{min} , height h_{fp} and width w_{fp} of the limits of the fingerprint. H and W are the height and width of the entire fingerprint image that includes the segmented fingerprint and the black background.

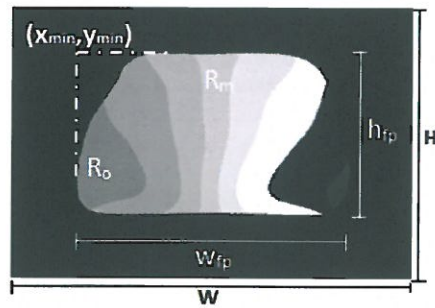


Figure 4.27: A PF DP containing the symbols used for the algorithm

To detect R_m , the area where all regions converge has to be located. To have a rough estimate of the row for the possible point of convergence, the outermost left region R_o is used as a guideline. Since all regions are moving closer to the point of maximum curvature, region R_o appears to be larger at a specific row. This row is denoted by Pt_c , as shown in Figure 4.28. The outermost region R_o , with a minimum row r_{min} and minimum col c_{min} , height h_{or} , and width w_{or} , is used to obtain the row of the converging point Pt_c . This is achieved by simply detecting the longest row Pt_c in region R_o .

On row Pt_c , the region with the smallest width w_{mreg} is searched for. The region with the smallest width will be referred to as R_m . The pixel values in region R_m is set to 255 and the other pixel values is set to 0. Region R_m is cropped from row Pt_c to the edge of the fingerprint. The cropped image is stored in R_{m2} . The maximum width of R_{m2} is denoted by w_{maxi} . Figure 4.29

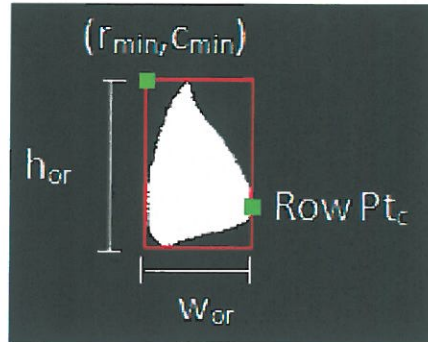


Figure 4.28: A *PF DP* containing row of Pt_c used for the algorithm

shows the width w_{mreg} and w_{maxi} that is obtained from region R_m and R_{m2} , respectively. The width w_{maxi} is compared to width w_{mreg} for classification. Algorithm 9 illustrates the procedure that needs to be followed to determine if the fingerprint is a *PA* or *PF*.

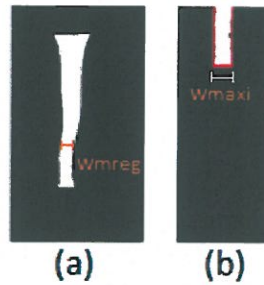


Figure 4.29: The widths used to classify a *PA* and a *PF* where (a) denotes the smallest width w_{mreg} in region R_m and (b) denotes the maximum width w_{maxi} in region R_{m2} (cropped R_m)

Algorithm 9 Classify a *DP* with no *SPs*

Input: P_{tc}, w_{fp}

Output: *class*

- 1: Find the region R_m by detecting the region smallest width w_{mreg} on row P_{tc}
 - 2: Set pixels in R_m to 255 and other pixels to 0.
 - 3: $height_C = H - r_c$
 - 4: // Crop R_m starting from row P_{tc} and
 - 5: // column 0 of width W and height $height_C$
 - 6: $R_{m2} = \text{Crop } R_m$
 - 7: Determine the maximum width w_{maxi} of R_{m2}
 - 8: **if** $(w_{mreg} - w_{maxi}) < 1/4 * w_{fp}$ **then**
 - 9: *class* = 'PF'
 - 10: **else**
 - 11: *class* = 'PA'
 - 12: **end if**
-

4.4 Testing

4.4.1 Experimental context

To answer the research question, experimental tests are conducted using the implemented classification scheme which uses *DPs* and *SPs* that are specifically designed to classify fingerprints with missing *SPs*. These experimental tests are conducted to evaluate the classification accuracy of the technique. This set-up exhibits the characteristics of a quantitative study that uses a scientific method.

The major limitation of a scientific method, is that it may not provide an accurate representation of whether the technique can perform in the real world due to external factors. In our context, the only external factor that directly affects the implemented algorithm, is the type of fingerprint output from the scanner which is used for the experiments. There are different types of fingerprint images namely: ink rolled fingerprints which have been scanned from paper; ink-rolled fingerprints on paper; ink flat fingerprints which have been scanned from paper; fingerprints which have been electronically rolled; and electronic flat fingerprints. To achieve a realistic set-up that meets the requirements of real world applications, the testing database must have the same type of images that experience the problem being solved (i.e. missing *SPs*). This is addressed in the next section.

4.4.2 Test data

Earlier classification techniques are often tested on the National Institute of Standard Technology (*NIST*) *DB*. This *DB* consists of ink rolled fingerprints on paper which have been scanned. The data contains complete fingerprints, some of which are rotated and have abundant noise, including text on images, smudges and background. Therefore, the central focus when performing classification on this *DB* is to create rotational and noise invariant techniques.

However, capturing techniques have evolved and fingerprints are often now captured electronically on small scanners. This results in fingerprint images being very different to those in the *NIST DB*. Therefore, the *NIST DB* does not meet the standards required to test algorithms specifically designed to solve challenges faced by current scanner images. These are complete, which makes it difficult to test the capabilities of the implemented algorithm that is specifically designed to handle fingerprints with missing *SPs*.

In 2000, the *FVC* introduced the largest publicly available *DB* that was specifically created to allow practitioners to test algorithms aimed at classifying scanner images. This allowed researchers to develop new software that satisfies the requirements of current technology. These databases are more appropriate for testing flat fingerprint classification. Therefore, the implemented algorithm will be tested on the *FVC 2002 DB1* and *FVC 2004 DB1*. Each *FVC DB* contains a total of 880 fingerprint samples [7].

4.4.3 Measurements

The main measure of classification is accuracy. The ideal way to represent the output, is by using a confusion matrix. The overall accuracy will be benchmarked using other rule-based techniques also focusing on missing *SPs*.

4.4.4 Software used for experimental testing

Matlab was used to implement the algorithm. This was chosen because the language has a toolbox for scientific image processing and analysis. This toolbox has good prototyping capabilities, making it much easier to detect bugs than a C++ and C. Furthermore, built in functions in Matlab require less scripting than other languages. In terms of speed, Matlab is much slower. However, the main measure of the algorithm's efficiency is accuracy rather than speed, particularly during this research phase. The algorithm could be transferred to another language if desired. Test results are shown in the next Chapter.

4.5 Conclusion

This chapter has described implementation prior to the classification algorithm, each fingerprint was rotated accordingly to produce unique patterns. A 3-region *DP* and a 24 *ND* search strategy was then used to detect the *SPs*. The rotation of the fingerprint based on the *SPs*, ensured it was invariant to rotation and unique patterns are obtained. The implementation for these stages was discussed fully in this chapter. The rules were derived from the unique properties of the *DP* with a specific a number of loops and deltas, namely: *W* with two loops/deltas; *RL*, *LL*, *TA* and *W* with single loop and single delta; *RL*, *LL* and *W* with single loop; *RL* and *LL* with single delta; and *PA* and *PF* cases.

Being able to develop rotational invariant classification rules, based on the uniqueness of *DPs* for flat fingerprint cases, highlights the potential of this algorithm in solving missing *SPs* challenges. To determine if *DPs* combined with *SPs* is truly successful in classifying flat fingerprints with missing *SPs*, the accuracy of the classification technique must be analysed. This will be covered in the next section.

Chapter 5

Experimental Results

In this chapter the classification accuracy of multiple re-implemented algorithms which accounted for flat fingerprint cases is compared to the proposed algorithm. The results are shown and discussed.

5.1 Introduction

This dissertation investigates whether flat fingerprints with missing *SPs* can be successfully classified by using *DPs* and the remaining *SPs*. A fingerprint classification algorithm using these features was presented to specifically address the issues with missing *SPs*. The algorithm was tested on electronically scanned images contained in *FVC 2002 DB1* and *FVC 2004 DB1*. To ensure a true interpretation of the level of accuracy, factors that contribute to the classification accuracy not achieving a 100% before even reaching the classification stage were removed. Therefore, 9 images from *FVC 2002 DB1* and 35 images from the *FVC 2004 DB1* were excluded. A total of 871 and 845 images were used from the *FVC 2002 DB1* and *FVC 2004 DB1*, respectively. The excluded images experienced problems related to the *SP* detection where there is loss of *SPs* on the *DPs*. Since these problems are related to *SP* detection and not classification, they should be excluded to test the classification alone. This was a small percentage of 0.03% of the entire set of images. An example of the excluded images are shown in Figure 5.1.

The ground truth of the class labels for each fingerprint in the *FVC DB* were manually created based on the definitions in [25]. The experimental results and a discussion based on the results are presented in this chapter.



Figure 5.1: Excluded fingerprints images due to loss of *SPs* on the *DPs*

5.2 Results

The images in the *DB* were grouped into seven classes namely: *RL*, *LL*, *TA*, *PA*, *PF*, and Unclassifiable classes *U*. An unclassifiable class *U* is a class of a specific type of flat fingerprint case where a no rule has been developed for it (i.e., a single delta *TA* and *W*).

Table 5.1: Confusion matrix of the classification using the proposed algorithm on 871 images from 880 images from the *FVC 2002 DB1*

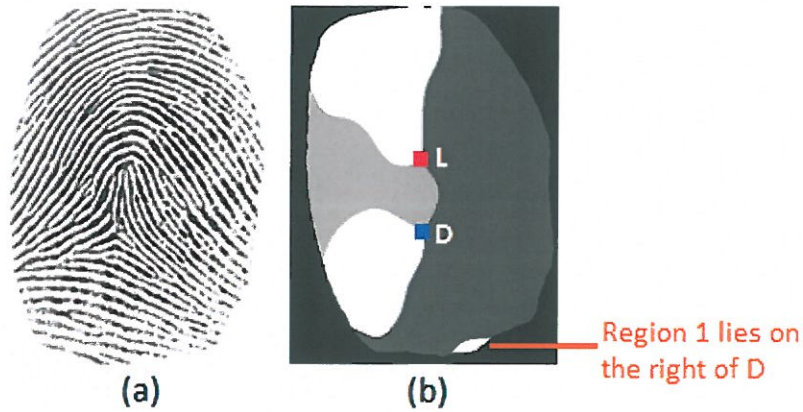
Assigned \ Actual	<i>W</i>	<i>RL</i>	<i>LL</i>	<i>TA</i>	<i>PA</i>	<i>PF</i>	<i>U</i>
<i>W</i>	185	7	2	0	0	0	0
<i>RL</i>	7	310	4	2	0	0	2
<i>LL</i>	10	4	288	0	0	0	2
<i>TA</i>	1	10	2	6	0	0	0
<i>PA</i>	0	1	1	0	8	6	0
<i>PF</i>	0	0	0	0	0	9	0
<i>U</i>	0	0	1	0	0	1	1
Accuracy	93%						

Tables 5.1 and 5.2 show the confusion matrices for the *FVC 2002* and *2004 DB1* that have excluded a total of 44 images. An example of the excluded images are shown in Figure 5.1. This provides a fair interpretation of the level of accuracy. The classification accuracy of 93% and 91.95% was achieved for *FVC 2002 DB1* and *FVC 2004 DB1*, respectively. Observing just these results alone, illustrates that using *DPs* and the remaining *SPs* found on the fingerprint can successfully classify fingerprints with missing *SPs*. The method produces a high classification accuracy since it addresses more cases of fingerprints with missing *SPs* and each flat fingerprint

Table 5.2: Confusion matrix of the classification using the proposed algorithm on 845 images from the 880 images from the *FVC 2004 DB1*

Assigned \ Actual	<i>W</i>	<i>RL</i>	<i>LL</i>	<i>TA</i>	<i>PA</i>	<i>PF</i>	<i>U</i>
<i>W</i>	183	18	9	0	0	0	0
<i>RL</i>	3	246	4	0	0	0	0
<i>LL</i>	13	2	283	0	0	0	0
<i>TA</i>	1	7	2	0	1	0	0
<i>PA</i>	0	0	1	0	52	0	0
<i>PF</i>	0	1	0	0	6	11	0
<i>U</i>	0	0	0	0	0	2	2
Accuracy	91.95%						

case are based on unique patterns. Furthermore the fingerprints were represented by a global pattern. These global patterns (*DPs*) eradicated the effect of noise on local orientation fields when minimal information is provided. The more successful classification cases occurred for classes *Ws*, *RLs*, *LLs*, and *PAs*.

**Figure 5.2:** *TA* class that produces a *DP* that resembles the characteristics of an *RL* [1]

Since the proposed rules have shown potential in classifying both complete and incomplete fingerprints, improving further on the proposed method would be most beneficial. In order to accomplish this, the downfalls of the method must be acknowledged. Therefore, by observing Table 5.1 and Table 5.2, it was found that errors occurred for complete *TA* classes which was misclassified has an *RL*. This error is common even in other classification algorithms techniques. An example of a *TA DP* classified as an *RL* is illustrated in Figure 5.2. In the segmentation process the orientation fields around the edges of the fingerprint experience distortion indicated in

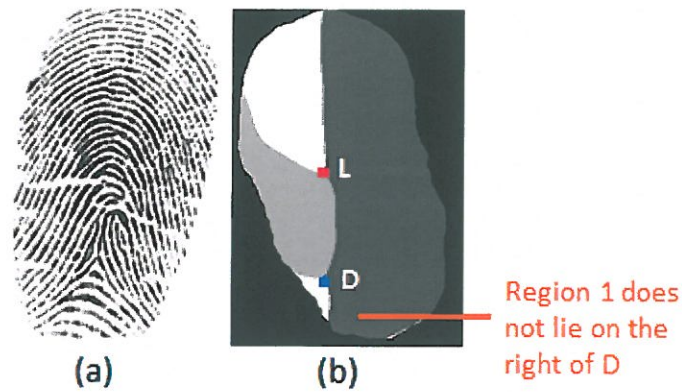


Figure 5.3: *RL* class that produces a *DP* that resembles the characteristics of a *TA*

Figure 5.2 by region 1 that lies on the right of the delta. Since the rule that classifies a *TA* and an *RL* is based on the width taken from the first occurrence of region 1 to the last occurrence of region 1 regardless if the region is not connected below the delta, it may be misclassified in such a case. A similar example which classified an *RL DP* as a *TA* is shown Figure 5.3. This error occurred due to the high similarities found in the structural layout of both fingerprints. The occurrence of these cases were still a small percentage since *TAs* rarely occur compared to *RLs*, *Ws* and *LLs*.

Table 5.3: Confusion matrix of the classification using the proposed algorithm on 880 images from the *FVC 2002 DB1*

Assigned \ Actual	<i>W</i>	<i>RL</i>	<i>LL</i>	<i>TA</i>	<i>PA</i>	<i>PF</i>	<i>U</i>
<i>W</i>	185	7	2	0	0	0	0
<i>RL</i>	9	311	4	2	0	0	2
<i>LL</i>	17	4	288	0	0	0	2
<i>TA</i>	1	10	2	6	1	0	0
<i>PA</i>	0	1	1	0	8	6	0
<i>PF</i>	0	0	0	0	0	9	0
<i>U</i>	0	0	1	0	0	1	1
Accuracy	91.93%						

Even when the test results included images shown in Figure 5.1 where the fingerprints' *DP* experienced *SP* loss, high accuracies of 91.93% and 87.50% are still obtained on the *FVC 2002 DB1* and *FVC 2004 DB1*, respectively. The results tested on the entire *DB* showing the actual classes against the output of the classification algorithm are shown in Figure 5.3 and Figure 5.4.

Table 5.4: Confusion matrix of the classification using the proposed algorithm on 880 images from the *FVC 2004 DB1*

Assigned \ Actual	<i>W</i>	<i>RL</i>	<i>LL</i>	<i>TA</i>	<i>PA</i>	<i>PF</i>	<i>U</i>
<i>W</i>	185	21	7	0	2	0	0
<i>RL</i>	3	252	4	0	2	0	0
<i>LL</i>	35	2	277	0	0	0	0
<i>TA</i>	1	7	2	0	5	0	0
<i>PA</i>	0	3	5	0	46	0	0
<i>PF</i>	0	1	4	0	5	9	0
<i>U</i>	0	0	1	0	0	1	1
Accuracy	87.50%						

The overall classification accuracy of 92.48% was benchmarked against previous algorithms which were designed to also classify cases of missing *SPs* using local orientation fields (but not *DPs*) and the remaining *SPs*. To directly compare the accuracy level of the proposed algorithm, methods by Karu and Jain, Msiza *et al.* and Webb and Mathekga have been re-implemented and tested on the same set of fingerprints. Table 5.5 show that the proposed algorithm outperforms several re-implemented methods presented in literature. The proposed method was also bench-

Table 5.5: Accuracy results of the proposed algorithm compared to algorithms in literature that were re-implemented

Author	<i>DB</i>	Average Accuracy	No. of classes
[56]	<i>FVC 2002 DB1, 2004 DB1</i>	50.53 % (867/1716)	5
[19]	<i>FVC 2002 DB1, 2004 DB1</i>	68.88 % (1182/1716)	5
[10]	<i>FVC 2002 DB1, 2004 DB1</i>	85.83% (1473/1716)	7
[7]	<i>FVC 2002 DB1, 2004 DB1</i>	90.67% (1556/1716)	7
Proposed Method	<i>FVC 2002 DB1, 2004 DB1</i>	92.48% (1587/1716)	7

marked against the work by Guo *et al.* and work by Jung and Lee. These works were not re-implemented since they were tested on similar *DBs* and have also excluded images. Jung and Lee tested on 650 out of 812 images and Guo *et al.* tested on 7345 out of 7920 images. The proposed method outperforms the Jung and Lee method by 12.38%. The improvement in accu-

Table 5.6: Accuracy results of the proposed algorithm compared to algorithms by Jung and Lee, and Guo *et al.*

Author	<i>DB</i>	Average Accuracy	No. of classes
[84]	<i>FVC 2002 DB1, 2004 DB1</i>	80.1 % (650/812)	4
[44]	<i>FVC 2002 DB1, 2004 DB1 & DB2</i>	92.7% (7345/7920)	4
Proposed Method	<i>FVC 2002 DB1, 2004 DB1</i>	92.48% (1587/1716)	7

racy is mainly due the formation of unique distinct global patterns formed for each fingerprint case and the fact that the algorithm could solve more cases than previous techniques. The proposed method is fairly close to the accuracy achieved by Guo *et al.* , with a difference of 0.22%. However, Guo *et al.* groups the *DB* into four classes by eliminating *U* and *PF*, and combined *PAs* and *TAs* into a single class. By doing this the accuracy of the algorithm will automatically increase as opposed to using a larger number of classes.

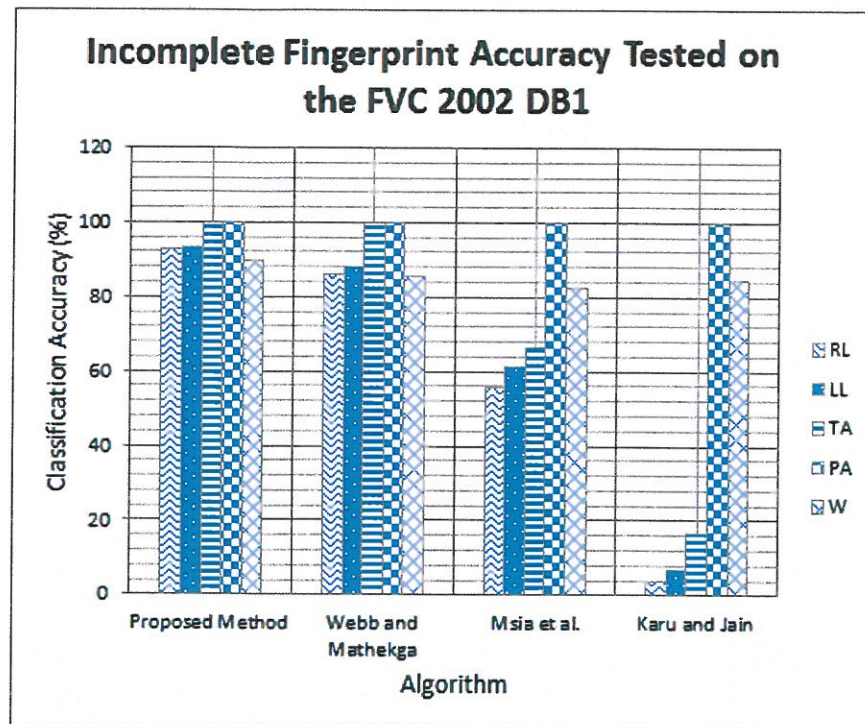


Figure 5.4: Accuracy achieve from imperfect fingerprint classes with missing *SPs* from the *FVC 2002 DB1*

Since the focus is on classification of fingerprints with missing *SPs*, it is more appropriate to observe how well the proposed method classifies only cases of fingerprints with missing *SPs*. Therefore, complete fingerprint classes were removed from the testing *DB*. In the *FVC 2004 DB1* there was a total of 134 *RLs*, 108 *LLs*, 0 *TAs*, 16 *PAs* and 168 *Ws* incomplete flat fingerprint cases. A larger number of incomplete flat fingerprint cases were found in the *FVC 2002 DB1* which consisted of a total of 167 *RLs*, 156 *LLs*, 6 *TAs*, 1 *PA*, and 186 *Ws*. Figure 5.4 and Figure 5.5 show the results of each incomplete fingerprint class (i.e., with missing *SPs*) of the proposed algorithm benchmarked against previous literature which have been re-implemented

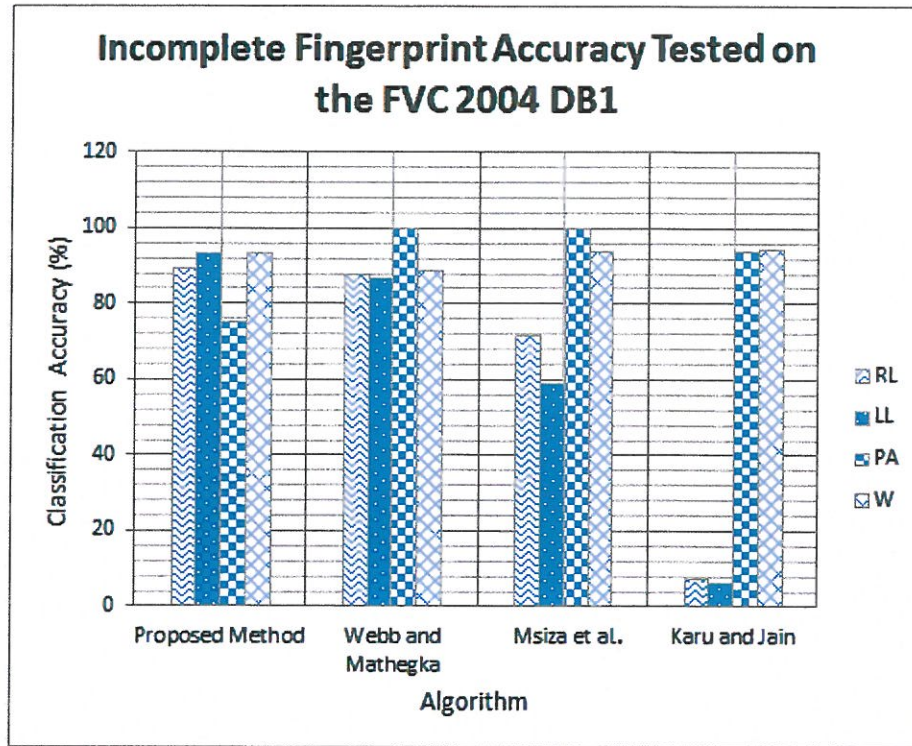


Figure 5.5: Accuracy achieve from imperfect fingerprint classes with missing *SPs* from the *FVC 2004 DB1*

and also attempts to classify missing *SPs*. The proposed method produced the highest accuracy on the *FVC 2002 DB* for incomplete fingerprints when compared to the result produced by Webb and Matheka [10], Msiza *et al.* [19], and Karu and Jain [56]. Even its lowest class accuracy result is a high 89.79 % (accuracy of incomplete *Ws*). The reason for the increased accuracy is owing to the robust set of rules that is based on global patterns. The algorithm also produced the highest accuracy for incomplete fingerprints of class *RL*, *LL*, and *W* on the *FVC 2004 DB1*. For the testing conducted on the *FVC 2004 DB1*, *PA* classes produced the lowest accuracy of 75%. The remaining 25% was identified to be a *PF*, which is ambiguous since an incomplete *PA* can be considered as a *PF* if the information below the highest curvature point is removed. Figure 5.6 and Figure 5.7 shows a complete *PA* and incomplete *PA* which have been successfully classified as a *PA* and a *PF*, respectively. This used the rules covered in Chapter 4.

The work of Msiza *et al.* and Karu and Jain have not developed a rule for *PF* cases. These methods classify a complete or incomplete fingerprint with no *SPs* as a *PA*. Owing to this their

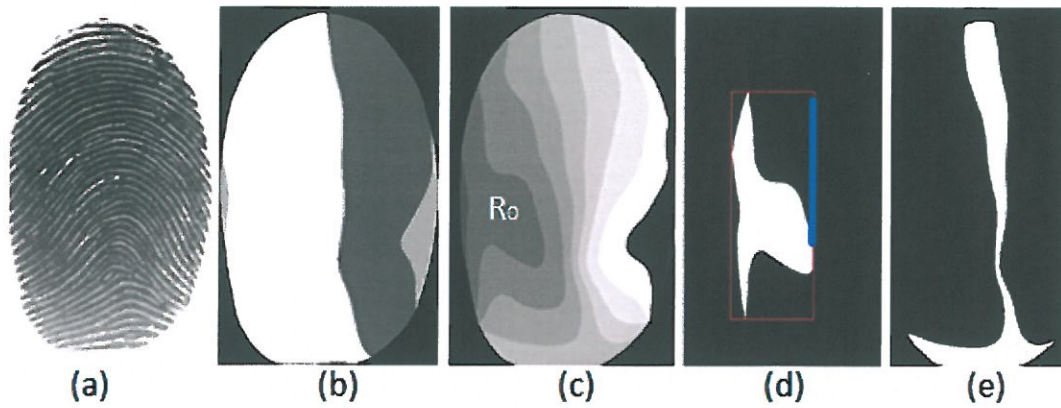


Figure 5.6: Successful classification of a *PA* where (a) is the segmented fingerprint, (b) 3-region *DP*, (c) 10-region *DP* (d) outer region and (e) mid region

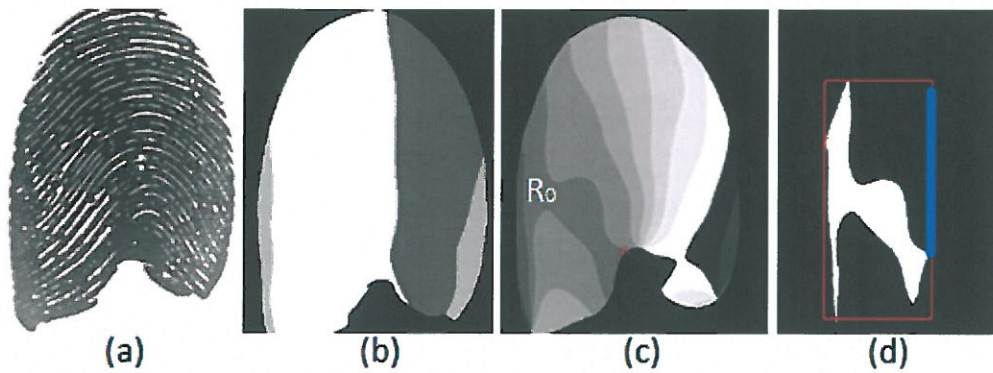


Figure 5.7: Successful classification of a *PF* where (a) is the segmented fingerprint, (b) 3-region *DP*, (c) 10-region *DP* and (d) outer region

accuracies are above 90%.

Table 5.7: Accuracy achieved for each fingerprint case using *FVC 2002 DB1*

Fingerprint case	Total	Number of correct cases	Accuracy
Two loops & two deltas	202	184	91.1
One loop (Top loop) & one delta	323	305	94.43
One delta	2	1	50
One loop (Top loop)	319	301	94.36
No SPs	21	16	76.19
One loop (Bottom loop)	4	3	75

Lastly, the accuracy for each individual fingerprint case which was tested on the *FVC 2002 DB1* and *FVC 2004 DB1* are shown in Table 5.7 and 5.8, respectively. Most flat fingerprint cases

Table 5.8: Accuracy achieved for each fingerprint case using the *FVC 2004 DB1*

Fingerprint case	Total	Number of correct cases	Accuracy (%)
Two loops & two deltas	208	189	90.87
One loop (top loop) & one delta	332	308	92.77
One delta	1	1	100
One loop (top loop)	246	227	92.28
No SPs	58	52	89.66
One loop (bottom loop)	0	-	-

produced a high average accuracy over 89%. However, there were too few fingerprints with a single delta, and *Ws* with a bottom loop to provide a fair interpretation of level of classification accuracy produced for these cases.

5.3 Conclusion

Experimental results of the proposed method produce a high classification accuracy when tested on fingerprints with missing *SPs*. Since each rule for each flat fingerprint case is based on unique and consistent *DPs*, it is less likely to result in misclassification. In addition, globally representing the fingerprints overcome the issues related to the uncertainty of local orientation especially in instances where there is minimum information captured.

Chapter 6

Conclusion

This chapter produces an overview of the conclusion of this research is based on the experimental results. Thereafter, suggestions for future work are discussed.

6.1 Summary

Can flat fingerprints with missing SPs be successfully classified by using the most appropriate fingerprint features? Indeed, this research showed that using the appropriate features is more capable of classifying fingerprints with missing SPs. Based on the advantages and disadvantages of each feature, DPs combined with SPs were concluded to be the more appropriate features. As a result of DPs being a global representation of a fingerprint, more information about the structural attributes of a class is provided [8]. This is beneficial for fingerprints with missing SPs since key structural properties are highlighted regardless of the amount of ridge information captured. Furthermore, the DPs of each flat fingerprint case were unique and consistent when the images were rotated based on SPs. This made it easier to develop simple classification rules which were invariant to rotation. The uniqueness and consistency of class patterns irrespective of the number of SPs detected, contributed to the classification's high average accuracy of 92.48% that was tested on the FVC 2002 DB1 and FVC 2004 DB1.

Like any novel rule-sets there will always be short comings and limitations. The most obvious one being higher misclassification received from an RL and TA with single loop and delta. This is caused by the distortion of orientation fields at the edge of the image after the segmentation process was applied.

Additionally, even though the bottom loop detection rule that classifies Ws did not contribute to misclassification, it was still limited to an extent. The reason behind why the rule could be

correctly classified is because the testing *DB* contained images that were rotated to maximum of 35° and the bottom loop detection rule catered for fingerprints rotated up till 50° . Furthermore, there were a small percentage of such cases occurring in the *FVC DBs*. However, this is the only rule that was not fully invariant to rotation. Further investigations can be conducted on a larger set of fingerprints with only bottom loops to establish a more robust rule.

Overall this study successfully showed that it can successfully classify fingerprints with missing *SPs* using features like *DPs* and the remaining *SPs* regardless of the limitation.

6.2 Future work

The ultimate goal of any work is to apply it to a real life application. This work shows potential. The following aspects are currently being improved to accomplish this:-

1. Further improvements can be made on the rule that classifies a *TA* and an *RL* with a single delta and single a loop.
2. Improvements can also be made by establishing a fully rotational invariant rule for a *W* with a bottom loop.
3. Real *DBs* often contain millions of fingerprints and therefore creating more classes is an option to decrease the search time. Based on the findings in this research, subclasses for complete *RLs*, *TAs* and *LLs* can be established using the *DPs*. For example an *RL* can form two sub-classes that are categorised as a 2-*CR* layout and a 3-*CR* layout. Investigations are being conducted to determine more subclasses for all complete fingerprints.
4. For many decades classification techniques have found it difficult to differentiate between a *PA* and a *PF*. The research finally introduces a rule that is more robust and dependent on the actual fingerprint pattern. Investigations can be undertaken to establish more properties that will indicate a *PF* from a *PA*. Using this knowledge improvements can be made on the rule and it can also aid in overcoming the limitations experienced by the fingerprint matching algorithms when a *PF* occurs.
5. Improvements can be made on the rule which classifies a single loop *DPs* with no regions below region 2.
6. Even though the rule for a single delta fingerprint is robust and invariant to rotation. It only classifies an *LL* and an *RL*, since no unique patterns were found for *Ws* and *TAs*.

However, further investigations can be conducted to develop rules for *Ws* and *TAs* with a single delta.

7. Further investigation can be made to identify subclasses for fingerprints with missing *SPs*.

References

- [1] K. R. Moses, P. Higgins, M. McCabe, S. Prabhakar, and S. Swann, "Automated fingerprint identification system (AFIS)," in *Sci. Work. Gr. Frict. Ridge Anal. Study Technol. Natl. Inst. Justice SWGFAST-The Fingerpr. Sourceb.*, 2011, pp. 1–33.
- [2] M. Liu, S. Liu, and Q. Zhao, "Fingerprint orientation field reconstruction by weighted discrete cosine transform," *Inf. Sci. (Ny)*, vol. 268, pp. 65–77, Jun. 2014. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0020025513005823>
- [3] R. Cappelli, D. Maio, and D. Maltoni, "Multi-classifier approach to fingerprint classification," *Pattern Anal. Appl.*, vol. 5, no. 2, pp. 136–144, 2002.
- [4] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni, "Fingerprint classification by directional image partitioning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 402–421, May 1999. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=765653>
- [5] L. Liu, C. Huang, and D. C. D. Hung, "Directional approach to fingerprint classification," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 22, no. 02, pp. 347–365, Mar. 2008. [Online]. Available: <http://www.worldscientific.com/doi/abs/10.1142/S0218001408006211>
- [6] G. Bahgat, A. Khalil, N. Abdel Kader, and S. Mashali, "Fast and accurate algorithm for core point detection in fingerprint images," *Egypt. Informatics J.*, vol. 14, no. 1, pp. 15–25, Mar. 2013. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S1110866513000030>
- [7] K. Dorasamy, L. Webb, J. Tapamo, and N. Khanyile, "Fingerprint classification using a simplified rule-set based on directional patterns and singularity features," in *8th IAPR Int. Conf. Biometrics*, vol. 2, Phuket, Thailand, 2015, pp. 400–407.

- [8] K. Dorasamy, L. Webb, and J. Tapamo, "Evaluation of the changes in directional patterns under the variation of rotation and number of regions," in *IEEE Int. Conf. Biometrics Spec. Interes. Gr.*, 2015, pp. 1–8.
- [9] G. Candela, P. Grother, C. Watson, R. Wilkinson, and C. Wilson, "PCASYS-A pattern-level classification automation system for fingerprints," *NIST Tech. Rep. NISTIR*, vol. 5647, 1995.
- [10] L. Webb and M. Mathekga, "Towards a complete rule-based classification approach for flat fingerprints," in *2014 Second Int. Symp. Comput. Netw.* South Africa, Pretoria: IEEE, Dec. 2014, pp. 549–555. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7052244>
- [11] Inboundtech.co, "Fingerprint Mcommerce: Mainstream for 2015," p. 1, 2015.
- [12] A. Curley, "Human Interface and Biometric Devices: Emerging Ecosystems, Opportunities and Forecasts 2014-2019," Tech. Rep., 2014.
- [13] B. Violino, "Reports Forecast Huge Growth for Mobile Biometric Security," p. 1, 2015. [Online]. Available: <https://securityintelligence.com/news/reports-forecast-huge-growth-mobile-biometric-security/>
- [14] A. K. Jain and U. Uludag, "Hiding biometric data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 11, pp. 1494–1498, 2003.
- [15] U. Uludag and A. K. Jain, "Attacks on Biometric Systems : A Case Study in Fingerprints," in *Proceeding SPIE-IS&T Electron. Imaging*, P. Wong and E. J. Delp III, Eds., vol. 5306, 2004, pp. 622–633.
- [16] A. Jain, L. Hong, and S. Pankanti, "Biometric identification," *Commun. ACM*, vol. 43, no. 2, pp. 91–98, 2000.
- [17] B. Miller, "Vital signs of identity," in *IEEE Spectr.*, no. February, 1994, pp. 22–30.
- [18] N. Yager and A. Amin, "Fingerprint classification: A review," *Pattern Anal. Appl.*, vol. 7, no. 1, pp. 77–93, Apr. 2004. [Online]. Available: <http://link.springer.com/10.1007/s10044-004-0204-7>

- [19] I. S. Msiza, B. Leke-Betechuoh, F. V. Nelwamondo, and N. Msimang, "A fingerprint pattern classification approach based on the coordinate geometry of singularities," in *Proc. 2009 IEEE Int. Conf. Syst. Man Cybern.* San Antonio, TX, USA: IEEE Computer Society, 2009, pp. 510–517.
- [20] L. O. Gorman, "Comparing passwords, tokens, and biometrics for user authentication," in *IEEE*, no. December, 2003, pp. 2021–2040.
- [21] J. A. Unar, C. W. Seng, and A. Abbasi, "A review of biometric technology along with trends and prospects," *Pattern Recognit.*, vol. 47, no. 8, pp. 2673–2688, 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.patcog.2014.01.016>
- [22] A. K. Jain, S. Prabhakar, and L. Hong, "A multichannel approach to fingerprint classification," *Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 4, pp. 348–359, 1999.
- [23] J. Nickell and J. Fischer, *Crime science: Methods of forensic detection*, 1999.
- [24] F. Galton, *Finger prints*. London and New York: Macmillan and Company, 1892.
- [25] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of fingerprint recognition*, 2nd ed. London: Springer, 2009.
- [26] W. Shen, M. Surette, and R. Khanna, "Evaluation of automated biometrics-based identification and verification systems," in *IEEE*, vol. 85, no. 9, 1997, pp. 1464–1478. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=628719>
- [27] J. Cowger, *Friction Ridge Skin: Comparison and Identification of Fingerprints*. CRC Press, 1992.
- [28] J. Hoover, "Fingerprinting: The work of the identification division of the Federal Bureau of Investigation," *Student Law*, vol. 13, 1961.
- [29] H. Lee, R. Ramotowski, and R. Gaensslen, *Advances in Fingerprint Technology*. CRC Press, 2001.
- [30] J. Chang and K. Fan, "A new model for fingerprint classification by ridge distribution sequences," *Pattern Recognit.*, vol. 35, no. 6, pp. 1209–1223, 2002.

- [31] C. Wilson, G. Candela, and C. Watson, "Neural network fingerprint classification," *J. Artif. Neural Netw.*, vol. 1, no. 2, pp. 203–228, 1994.
- [32] J. Min and S. Cho, "Multiple Decision Templates With Adaptive Features for Fingerprint Classification," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 21, no. 08, pp. 1323–1338, Dec. 2007. [Online]. Available: <http://www.worldscientific.com/doi/abs/10.1142/S0218001407006009>
- [33] M. Liu, "Fingerprint classification based on adaboost learning from singularity features," *Pattern Recognit.*, vol. 43, no. 3, pp. 1062–1070, 2010. [Online]. Available: <http://dx.doi.org/10.1016/j.patcog.2009.08.011>
- [34] K. Cao, L. Pang, J. Liang, and J. Tian, "Fingerprint classification by a hierarchical classifier," in *Pattern Recognit.*, vol. 46, no. 12, Dec. 2013, pp. 3186–3197. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0031320313002124>
- [35] A. Senior, "Markov model fingerprint classifier," in *Conf. Rec. Thirty-First Asilomar Conf. Signals, Syst. amp.*, no. 4000. IEEE, 1997, pp. 306–310.
- [36] M. Chong, T. Han Ngee, L. Jun, and R. K. Gay, "Geometric framework for fingerprint image classification," *Pattern Recognit.*, vol. 30, no. 9, pp. 1475–1488, 1997.
- [37] A. P. Fitz and R. J. Green, "Fingerprint classification using a hexagonal fast fourier transform," *Pattern Recognit.*, vol. 29, no. 10, pp. 1587–1597, 1996.
- [38] M. Liu, "Phase harmonic transforms to fingerprint classification," in *Emerg. Top. Comput. Vis. it's Appl.* USA: World Scientific, 2011, ch. 1, pp. 1–17.
- [39] J. Luo, D. Song, C. Xiu, S. Geng, and T. Dong, "Fingerprint classification combining Curvelet transform and gray-level co-occurrence matrix," *Math. Probl. Eng.*, 2014.
- [40] G. C. Zacharias and S. P. Lal, "Singularity detection in fingerprint image using orientation consistency," in *2013 Int. Multi-Conference Autom. Comput. Commun. Control Compress. Sens.* India: IEEE, 2013, pp. 150–154.
- [41] A. Tiganasu and C. Lazar, "Singular regions detection in fingerprint images," in *17th Int. Conf. Syst. Theory, Control Comput.* IEEE, 2013, pp. 551–556.

- [42] A. M. Bazen and S. H. Gerez, "Systematic methods for the computation of the directional fields and singular points of fingerprints," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 905–919, Jul. 2002. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1017618>
- [43] L. Hong, Y. Wan, and A. Jain, "Fingerprint image enhancement : algorithm and performance evaluation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 777–789, 1998.
- [44] J. Guo, Y. Liu, J. Chang, and J. Lee, "Fingerprint classification based on decision tree from singular points and orientation field," *Expert Syst. Appl.*, vol. 41, no. 2, pp. 752–764, 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.eswa.2013.07.099>
- [45] M. Liu, "Fingerprint classification based on singularities," in *Chinese Conf. pattern Recognit.* IEEE, 2009, pp. 1–5.
- [46] B. Cho, J. Kim, J. Bae, I. Bae, and K. Yoo, "Core-based fingerprint image classification," in *Proc. 15th Int. Pattern Recognit.* Korea: IEEE, 2000, pp. 859–862.
- [47] M. Kawagoe and A. Tojo, "Fingerprint pattern classification," *Pattern Recognit.*, vol. 17, no. 3, pp. 295–303, 1984.
- [48] J. Hong, J. Min, U. Cho, and S. Cho, "Fingerprint classification using one-vs-all support vector machines dynamically ordered with naive bayes classifiers," *Pattern Recognit.*, vol. 41, no. 2, pp. 662–671, Feb. 2008. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0031320307003299>
- [49] S. Javed and A. Usman, "Computerized system for fingerprint classification using singular points," in *14th Int. Multitopic Conf.* IEEE, 2011, pp. 96 – 101.
- [50] X. Wang, F. Wang, J. Fan, and J. Wang, "Fingerprint classification based on continuous orientation field and singular points," in *Int. Conf. Intell. Comput. Intell. Syst.* China: IEEE, 2009, pp. 189–193.
- [51] D. Batra, G. Singhal, and S. Chaudhury, "Gabor filter based fingerprint classification using support vector machines," in *Proc. 1st IEEE India Annu. Conf. - INDICON.* India: IEEE,

- 2004, pp. 256–261. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1497751>
- [52] Y. Yao, L. G. Marcialis, M. Pontil, P. Frasconi, and F. Roli, “Combining flat and structured representations for fingerprint classification with recursive neural networks and support vector machines,” *Pattern Recognit.*, vol. 36, no. 2, pp. 397–406, 2003.
- [53] D. Maio and D. Maltoni, “A structural approach to fingerprint classification,” in *Proc. 13th Int. Conf. pattern Recognit.* Italy: IEEE, 1996, pp. 578–585.
- [54] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, “FVC2000 : Fingerprint Verification Competition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 402–412, 2002.
- [55] H. Jung and J. Lee, “Noisy and incomplete fingerprint classification using local ridge distribution models,” *Pattern Recognit.*, vol. 48, no. 2, pp. 473–484, Feb. 2015. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0031320314002891>
- [56] K. Karu and A. K. Jain, “Fingerprint classification,” *Pattern Recognit.*, vol. 29, no. 3, pp. 389–404, 1996.
- [57] D. Maio, D. Maltoni, R. Cappelli, J. Wayman, and A. Jain, “FVC2002: Second Fingerprint Verification Competition,” 2002.
- [58] A. K. Jain, S. Prabhakar, and P. Sharath, “Matching and fingerprint classification-A case study on fingerprint domain,” in *Proceedings- Indian Natl. Sci. Acad. Part A 67.2*, 2001, pp. 223–242.
- [59] I. S. Msiza, J. Mistry, B. Leke-Betechuoh, F. V. Nelwamondo, and T. Marwala, “On the introduction of secondary fingerprint classification,” in *State art biometrics*, J. Yang and L. Nanni, Eds. China, Europe: InTech, 2009, ch. 5, pp. 104–120. [Online]. Available: www.intechopen.com/books/state-of-the-art-in-biometrics/on-the-introduction-of-secondary-fingerprint-classification
- [60] S. Jirachaweng, Z. Hou, W. Yau, and V. Areekul, “Residual orientation modeling for fingerprint enhancement and singular point detection,” *Pattern Recognit.*, vol. 44, no. 2,

- pp. 431–442, Feb. 2011. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S003132031000405X>
- [61] N. Nain, B. Bhadviya, B. Gautam, and D. Kumar, “A fast fingerprint classification algorithm by tracing ridge flow patterns,” in *Int. Conf. Signal Image Technol. Internet Based Syst.* IEEE, 2008, pp. 235–238.
- [62] M. Mostafa, A. Allah, M. M. Ali, and A. I. Ali, “A new fingerprint classification approach based on fast fourier transformer,” in *Proceeding 5th Int. Conf. Informatics Syst.*, 2008, pp. 78–83.
- [63] C. Singh and A. Kaur, “Fast computation of polar harmonic transforms,” *J. Real - Time Image Process.*, vol. 10, no. 1, pp. 59–66, 2012.
- [64] S. Wang, W. W. Zhang, and Y. S. Wang, “Fingerprint classification by directional fields,” in *Proc. fourth IEEE Int. Conf. multimodal interfaces*, no. 5. IEEE Comput. Soc, 2002, pp. 395–399. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1167027>
- [65] V. Conti, C. Militello, S. Vitabile, and F. Sorbello, “Introducing pseudo-singularity points for efficient fingerprints classification and recognition,” in *2010 Int. Conf. Complex, Intell. Softw. Intensive Syst.* IEEE, 2010, pp. 368–375.
- [66] A. Tariq, M. U. Akram, and S. A. Khan, “An automated system for fingerprint classification using singular points for biometric security,” in *6th Int. Conf. Internet Technol. Secur. Trans.*, no. December. IEEE, 2011, pp. 11–14.
- [67] A. K. Nagaty, “Fingerprint classification using artificial neural networks: A combined structural and statistical approach,” *Neural Networks*, vol. 14, no. 9, pp. 1293–1305, 2001.
- [68] Y. A. Wang and J. Hu, “Global ridge orientation modeling for partial fingerprint identification,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 72–87, Jan. 2011. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/21088320>
- [69] M. Kaas and A. Witkin, “Analysing oriented patterns,” *Comput. Vision, Graph. Image Process.*, vol. 37, no. 3, pp. 362–385, 1987.

- [70] L. Ji and Z. Yi, "SVM-based fingerprint classification using orientation field," in *3rd Int. Conf. Nat. Comput.*, no. Icnc. IEEE, 2007, pp. 7–10.
- [71] D. Maio and L. Nanni, "A two-stage fingerprint classification system," in *Proceeding 2003 ACMSIGMM Work. Biometrics methods Appl.*, 2003, pp. 95–99.
- [72] M. Galar, J. Derrac, D. Peralta, I. Triguero, D. Paternain, C. Lopez-Molina, S. García, J. M. Benítez, M. Pagola, E. Barrenechea, H. Bustince, and F. Herrera, "A survey of fingerprint classification Part I: Taxonomies on feature extraction methods and learning models," *Knowledge-Based Syst.*, vol. 81, pp. 76–97, Jun. 2015. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0950705115000519>
- [73] A. Erol and U. Halici, "The effect of orientation map accuracy on fingerprint classification using PCA-MSO," in *Fourth Int. Conf. knowledge-based intelligent Eng. Syst. allied technology*. Brighton, UK: IEEE, 2000, pp. 2–5.
- [74] J. Kajang-puchong, "The use of SOM for fingerprint classification," in *Int. Conf. Inf. Retr. Knowl. Manag.*, no. 1, 2010, pp. 287–290.
- [75] G. P. O. B. Hobart, "Fingerprint classification using singular points and their principal axes," in *2004 Int. Conf. Image Process.*, 2004, pp. 0–3.
- [76] D. Weng, Y. Yin, and D. Yang, "Singular points detection based on multi-resolution in fingerprint images," *Neurocomputing*, vol. 74, no. 17, pp. 3376–3388, Oct. 2011. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0925231211003596>
- [77] S. Bernardlv, N. Boujemaa, D. Vitale, and C. Bricot, "Fingerprint classification using kohonen topologic map," in *Int. Conf. Image Process.* IEEE, 2001, pp. 230–233.
- [78] L. Hong and A. K. Jain, "Classification of fingerprint images," in *Proc. Scand. Conf. Image Anal.*, East Lansing, 1999, pp. 665–672.
- [79] Q. Zhang, K. Huang, and H. Yan, "Fingerprint classification based on extraction and analysis of singularities and pseudoridges," *Pattern Recognit.*, vol. 37, no. 11, pp. 2233–2243, 2004.

- [80] Y. Choi and G. Kim, "Graph-based fingerprint classification using orientation field in core area," *IEICE Electron. Express*, vol. 7, no. 17, pp. 1303–1309, 2010. [Online]. Available: <http://joi.jlc.jst.go.jp/JST.JSTAGE/elex/7.1303?from=CrossRef>
- [81] Y. Mei, R. Hou, and J. Wang, "An improved method for fingerprints singular points detection based on orientation field partition," *Int. J. Signal Process. Process. Pattern Recognit.*, vol. 6, no. 1, pp. 225–234, 2013.
- [82] L. Wang, N. Bhattacharjee, and B. Srinivasan, "Fingerprint reference point detection based on local ridge orientation patterns of fingerprints," in *WCCI 2012 IEEE World Congr. Comput. Intell.* Brisbans, Australia: IEEE, 2012, pp. 10–15.
- [83] L. Wang, N. Bhattacharjee, G. Gupta, and B. Srinivasan, "Adaptive approach to fingerprint image enhancement," in *Proc. 8th Int. Conf. Adv. Mob. Comput. Multimed.*, 2010, pp. 42–49.
- [84] H. Jung and J. Lee, "Fingerprint classification using the stochastic approach of ridge direction information," in *Int. Conf. fuzzy Syst.* IEEE, Aug. 2009, pp. 169–174. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5277309>