Fingerprint processing based on phase portrait model

Li, Jun

2006

Li, J. (2006). Fingerprint processing based on phase portrait model. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/4645>

<https://doi.org/10.32657/10356/4645>

Nanyang Technological University

Downloaded on 26 Apr 2023 17:12:21 SGT

**Fingerprint Processing
Based On
Phase Portrait Model**

LI Jun

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy

2006

*To my Family,
for their encouragement and love.*

Acknowledgements

My first and most earnest acknowledgment must go to my supervisor: Dr. Yau Wei Yun who is a truly remarkable advisor. He grants me a lot of freedom to explore new ideas, but at the same time interacts closely with me. For the past 4 years, I have benefited enormously from his breadth of knowledge, capacity for insight and invaluable advice and guidance. In every sense, none of this work would have been possible without him. I should also show my biggest debt of gratitude to Dr. Yau for his encouragement, support, care in all aspects of my Ph.D life. What he had done for me will be kept in my deep heart and be cherished forever.

I also owe a huge debt of gratitude to my supervisor: Professor Wang Han. Though it is for two years that I am under Prof. Wang's guidance, I benefit a lot from his suggestion and his help. "Work hard and play hard", what Prof. Wang said to me, will be together with me.

I am indebted to Goh Seng Lee for his great help of my Ph.D work, Dr. Toh Kar Ann for his suggestion, Ms. Liu Manhua for her valuable discussion.

Far too many people to mention individually have assisted in so many ways during my stay at NTU and I²R. They all have my sincere gratitude. In particular, I would like to thank all members in the biomedical instrumentation Lab, Liu Sheng, Tang Wenyin, Wu Xingye, Wu Yilei, Zhang Dingguo, Zhang Runxuan, Zhao Peng, Zhou

Xin, Zhou Hao, Xia Yang, Yan Min, Chen Xiaofang, Chen Ming, Liang Hongzhu and Yang Xin, Yap Sing Loon. I also would like to thank all members in the MMB Lab.: Lawrence Chen, Lam Hwee Keong, Wang Hee Lin, Wang Kejian, Wang Jiang Gang, Tang Ka Yin, Eng How Lung for the wonderful time we have spent.

I would also like to thank Nanyang Technological University for the much needed financial support that enables me to complete my PhD programme.

A penultimate thank-you goes to my beloved parents and brothers. For always being there when I needed them most, and never once complaining about how infrequently I visit, they deserve far more credit than I can ever give them.

My final and most heartfelt acknowledgment must go to Chien YuYi, who has accompanied me for these four years. I have promised that this work is dedicated to her. Now it was the time to thank for her love and those happy time we spent together.

Summary

In recent decades, automatic fingerprint identification system has attracted significant interest among the scientific research community. Though large amount of effort has been made in both the academic and industrial communities, there are still unsolved problems. Noisy image processing and large scale database processing are two typical problems.

Fingerprint images consist of oriented texture. Therefore the orientation information plays a very important role in fingerprint processing. Moreover, the orientation of the fingerprint follows a certain structure that is not random. Such oriented structure provides a possibility to recover the orientation information in fingerprints corrupted by noise, even for a large noise patch which cannot be solved by the traditional gradient based methods. Accurate reconstruction of the orientations is useful for fingerprint filtering, segmentation, classification and recognition.

Several methods have been proposed to reconstruct the fingerprint orientation. In this thesis, phase portrait model approach is used. The first order phase portrait is demonstrated to be robust for both occluded and noisy fingerprints. It also has the capability to describe the orientation patterns of the core point and delta point in a fingerprint image. Therefore a method based on the first order phase portrait is proposed to reconstruct the local orientation near the core and delta points. Fur-

thermore, considering the regularity of the fingerprint orientation and linearity of the first order phase portrait, a model based on the piecewise phase portrait is used to predict the orientations far away from the singular point region. The advantage of this prediction model is that it provides a way to overcome the large patch of noise or missing orientation. To obtain the fingerprint orientation with higher precision, more complicated model based on the constrained nonlinear phase portrait approach is proposed. The model employs the advantages of both the first order and high order phase portrait model. Hence accurate reconstructed orientations at both the local and global area can be obtained simultaneously.

This thesis also addresses fingerprint classification using the coefficients of the constrained nonlinear phase portrait model together with the singularities information. Since the coefficients of the constrained nonlinear phase portrait model play a key role in reconstructing the fingerprint orientation, it is also expected to be able to effectively classify fingerprints. A fingerprint classification algorithm using the orientation model and the singular points' information which complement the orientation information is developed and is shown to have good classification performance.

Contents

Acknowledgements	i
Summary	iii
List of Figures	x
List of Tables	xiii
1 Introduction	1
1.1 Motivation	2
1.2 Objectives	5
1.3 Major Contributions of the Thesis	6
1.4 Organization of the Thesis	7
2 Literature Review	8
2.1 Fingerprints as Biometrics	8
2.1.1 Overview of General Fingerprint Processing and the Influence of Orientation	8
2.1.2 General Methods to Deal With the Noisy Images	12

2.1.3	Prior Works Related to Partial Image Processing	13
2.2	Phase Portrait Model Overview	14
2.2.1	Introduction	15
2.2.2	Linear Phase Portrait Model	16
2.2.3	Nonlinear Phase Portrait Model	18
2.2.4	Orientation Description Using Phase Portrait Model	19
3	Singular Point Detection Based On Phase Portrait	23
3.1	Literature Review of Singular Point Detection in Fingerprint Processing	24
3.1.1	Poincare Index-Based Methods	25
3.1.2	Intersection-Based Methods	27
3.1.3	Filter-Based Methods	28
3.2	Singular Point Detection Based On Phase Portrait	30
3.2.1	Theoretical Background	30
3.2.1.1	Robustness Analysis	34
3.2.2	Prior Works on General Singular Point Detection Using Phase Portrait	37
3.2.2.1	Isotangent-Based Algorithm	38
3.2.2.2	Direct Estimation	39
3.3	General Procedure of the Proposed Method	40
3.4	Experiments	44
3.5	Summary	49
4	Prediction Model Based On The First Order Phase Portrait	51

4.1	Introduction	51
4.2	Literature Review on Fingerprint Orientation Modeling	54
4.3	Orientation Behavior Analysis of Fingerprints	58
4.3.1	Default Global Orientation Model	58
4.3.2	Orientation Behavior Near Singular Points	60
4.3.2.1	The First Order Phase Portrait Model	61
4.3.2.2	Piecewise Phase Portrait Model	62
4.4	Prediction Model Applied to the Complete Input	65
4.4.1	Criteria for Piecewise Model	66
4.4.2	Rectification of Prediction Model	68
4.4.2.1	Discontinuity Removal	68
4.4.2.2	Weighted Piecewise Models	69
4.5	Prediction Model Applied to the Partial Input	75
4.5.1	Search Model for the Incomplete Image	75
4.5.1.1	Distance Measure	75
4.5.1.2	Searching the Best Fit Model	76
4.5.1.3	Speeding Up the Search Algorithm	80
4.6	Experiments	82
4.6.1	Prediction Model Applied to the Complete Input	82
4.6.2	Prediction Model Applied to Partial Input	86
4.7	Summary	90
5	Constrained Nonlinear Phase Portrait Model	93
5.1	Introduction	93

5.2	Constrained Nonlinear Phase Portrait Model	94
5.3	Solution of Coefficients and Perturbation Analysis	97
5.3.1	Robustness in Case of a Large Patch of Noise	101
5.3.2	Robustness in Cases where Singular Points Are Close to One Another	102
5.4	Implementation of the Proposed Model	105
5.5	Experiments	109
5.6	Summary	114
6	Fingerprint Classification Based On Nonlinear Phase Portrait Model	117
6.1	Introduction	117
6.2	Literature Review on Fingerprint Classification	122
6.3	Feature Extraction	127
6.3.1	Relationship between Detection of the Singular Points and Computation of Orientation Image	128
6.3.2	Complex Symmetrical Filters Based Method to Detect the Candidate Singular Points	130
6.3.3	Constrained Nonlinear Phase Portrait Model of Fingerprint Orientation	132
6.3.4	Interactive Validations of Features	133
6.3.5	Form the Feature Vectors for Fingerprint Classification	136
6.4	Implementation of the Proposed Method	137
6.5	Experimental Results	146
6.5.1	Dataset	146

6.5.2	Performance Measurement	147
6.5.3	Coefficients Used for Classification	148
6.5.4	Classification Results Using Orientation Model	151
6.5.4.1	NN as the Classifier	151
6.5.4.2	SVM as the Classifier	153
6.5.4.3	RM Model as the Classifier	153
6.6	Summary	157
7	Conclusions And Future Works	161
7.1	Conclusion	161
7.2	Future Work	163
	Appendix	167
A	Reconstruction Of The Delta Pattern From Two Arbitrary Asymptotes	167
	Author's Publications	171
	Bibliography	173

List of Figures

3.1	Singular points in a fingerprint image	24
3.2	An example for localization of singular points using transition lines .	27
3.3	An example of the focal point detection approach	29
3.4	Patterns and the corresponding complex filters	29
3.5	The parameters estimated from orientation images with Gaussian noise of variance from 0 radian to 1.5 radian	35
3.6	Robustness analysis of phase portrait with noise input	36
3.7	Robustness analysis of phase portrait with occluded data	37
3.8	Distance from the estimated to true singular point using occluded orientation images with Gaussian noise of variance from 0 radian to 0.3 radian	37
3.9	Behaviors of singular points	42
3.10	Intermediate results of SP detection using phase portrait for loop . .	45
3.11	Intermediate results of SP detection using phase portrait for whorl . .	47
3.12	Examples of SP detection using phase portrait	48
3.13	The comparative examples of SP detection	49
4.1	Examples of FBI 8-class scheme	53

4.2	Examples of zero-pole based model	56
4.3	Default global orientation behavior	58
4.4	Default global orientation behavior with varying value of parameters a and b	59
4.5	Behaviors of singular points	62
4.6	Orientation behaviors near singular points	63
4.7	Examples of the piecewise prediction model	67
4.8	Weighted models for the arch class	71
4.9	Weighted models near the delta point of the loop	72
4.10	Examples before and after the rectification	73
4.10	Examples before and after the rectification(Con'd)	74
4.11	Ambiguity of orientation image	76
4.12	Division of the whorl fingerprint type	80
4.13	Possible positions of singular points in predefined models	81
4.14	Examples of prediction algorithm	84
4.15	Sample performance of the prediction algorithm.	85
4.16	Examples of the search model	89
4.17	Error analysis for search algorithm	91
5.1	Examples of fingerprint with two close singular points	102
5.2	Reconstructed orientation using Eqns. 5.18	104
5.3	Resultant images obtained in each step	108
5.4	Examples of reconstructed orientation	111

5.5	Comparative results between the proposed model and the combination model	112
5.6	Examples of the reconstructed orientation of poor quality image using the proposed model	113
5.7	Example of a poor quality image where the reconstruction fails	114
6.1	Examples of five-class scheme	119
6.2	Examples of small-interclass variability and large-intraclass variability	120
6.3	Examples of noisy images	121
6.4	An example for the interactive validation of the singular points	135
6.5	Results for each step in the preprocessing stage	139
6.6	Examples of the angles between the primary core point and the delta points	143
6.7	An example of an ambiguous fingerprint class	147
6.8	Reconstructed orientation using the different order coefficients	150
6.9	Graph of error rate vs. number of hidden nodes	152
A.1	Examples of the delta pattern with different A	168
A.2	Explanation of the reconstruction using the two asymptotes	169

List of Tables

2.1	Classes of the first order phase portrait	17
3.1	Comparison of the accuracy of singular point detection	45
3.2	Computational cost comparison of the singular point detection algorithms	49
4.1	Pseudo-code for search algorithm	77
4.2	Overall results of the Prediction model	86
4.3	Results of search algorithm from 800 images used in the experiment .	90
5.1	Comparison results of the different orientation models	109
6.1	FBI fingerprint classification scheme	118
6.2	Classification based on singularities	122
6.3	Classification results with different amounts of coefficients	148
6.4	Confusion matrix with NN as the classifier	153
6.5	Confusion matrix with SVM as the classifier	154
6.6	Classification results by RM	155
6.7	Confusion matrix with RM as the classifier	156

6.8	Confusion matrix with NN as the classifier by the Jain's fingercode method [1]	157
6.9	Confusion matrix with SVM as the classifier by both the fingercode and RNN-extracted features [2]	157
6.10	Confusion matrix by the Cappelli's method [3]	157
6.11	Confusion matrix by the Zhang's method [4]	158
6.12	Literature reviews of fingerprint classification	160

Chapter 1

Introduction

With the rapid development of information technology and the growing importance of electronic transactions in the commercial and service industries, identity authentication together with information security has become increasingly important. Traditional methods for identity authentication such as passwords or ID cards have an obvious disadvantage: the person who offers the password or ID card cannot prove that he or she is who he or she claims to be. If one loses his card or password, the others who get his card or password could obtain the same private rights as he or she owns. Biometric technology can potentially solve such problem. Biometrics refers to the use of distinctive physiological (e.g. fingerprints, face, retina, iris) and behavioral (e.g. gait, signature) characteristics to automatically identify individuals. Compared with passwords and ID cards, biometrics cannot be stolen, lost, forgotten or borrowed. These inherent attributes make biometrics as an effective complement to secure identity authentication.

In recent years, especially after the 9.11 incident, automatic biometric personal identification has become a popular research topic. As the oldest biometric trait, a fingerprint is made of a series of ridges and furrows on the surface of the finger.

Sir William Herschel Galton showed in 1888¹ through a large scale study that no two fingerprints are exactly the same and that prints remain the same throughout a person's lifetime . It is the basis for fingerprint identification and is widely accepted today. Fingerprints have been widely used in forensic, commercial and civilian applications due to its stability, acceptability and high accuracy. Today the availability of smaller and cheaper solid-state fingerprint sensors broadens the application of automatic fingerprint identification system to even the consumer products.

Though many efforts have been made, automatic personal identification using fingerprints is still a challenge to researchers. The typical problems include how to deal with the noisy image and how to reduce the processing time in the huge database. Hence, there are still research topics to be investigated before automatic fingerprint identification systems become truly pervasive.

1.1 Motivation

A fingerprint image consists of a series of oriented texture. This implies that fingerprint orientation is very important when processing a fingerprint image. In the following part, we list some general topics of fingerprints processing and the role of orientation in these topics.

1. Fingerprint image enhancement: A fingerprint image acquired from a sensor always contains noise due to the skin condition (dry or moist), sensor noise, dust, scar etc.. Therefore the extraction of useful information for identification becomes difficult. The goal of fingerprint image enhancement is to improve the clarity of the ridge structure. A typical fingerprint enhancement algorithm is

¹<http://www.fingerprintamerica.com/fingerprintHistory.asp>

based on contextual filters, in which the context is often defined by the **local ridge orientation** and local ridge frequency.

2. Fingerprint image segmentation: Fingerprint segmentation is to select the ridge-furrow structure (or foreground information) from the non ridge-furrow structure (or background information). Undoubtedly, as an orientated texture, **clear orientation** is critical to discriminate a fingerprint image from its background. Other useful information may include variance and mean of the gray level image, local frequency etc..
3. Feature extraction: The most popular feature extracted from the fingerprint image is minutia. To locate these minutiae, we can trace the ridges, in which the **orientation** is needed. Furthermore, the feature vector for one minutia contains the position and the **orientation** of the minutia [5].
4. Fingerprint classification: As coarse level information, fingerprint orientation can be used to classify fingerprints as well (more details please refer to Chapter 6).

In all these topics, orientation pattern plays an important role. It can provide additional useful information. However there are many fingerprint images that are noisy, caused by dust, oil, moisture, scars or excessively wet or dry fingers. As such, we cannot always obtain correct orientation patterns. This is especially the case for forensics application. For other applications, where possible, recapturing the fingerprint to obtain better quality version is preferred. However, this may not be possible all the time such as when the sensor surface is dirty, degraded or damaged after many touches. In addition, there are instances of fingers which are inherently of poor quality. According to Hong and Jain [6], about 10% of the fingerprint images are of poor quality, especially the fingers of elderly people and manual workers. For

these fingers, re-capturing the fingerprints would not improve the quality further. In such cases, the solution commonly used is to perform filtering. However, filtering only operates in a local region and it cannot solve large noisy or missing patches in the image.

In this work, we address global modeling of the fingerprint orientation image. “Global modeling” means that we reconstruct the orientation image not only considering the neighborhood orientations but also the whole orientation trend. The advantages of modeling fingerprint orientation images are:

1. Able to acquire accurate orientation pattern: the orientations computed from the real input image are not always accurate due to noise. Orientation model can give a mathematical description according to the whole orientation tendency. Thus, it is robust to noise and able to withstand missing or noisy information of larger proportion.
2. Able to describe all the orientation patterns in one concise and unified form. Prior work in describing the orientation pattern depends on the accurate detection of all singular points in the fingerprint [7,8]. However, in practice, it is not always the case that all singular points can be detected or available. Instead, our approach is more tolerant to the availability of accurate singular points. It also provides a concise and unified form to reconstruct the orientation.
3. Able to use the additional information to classify fingerprints into distinct classes. We can use the parameters obtained from the orientation model to classify the fingerprints into several sub-classes in large databases for quick search.
4. Assist in generating artificial fingerprint images. A good fingerprint recognition algorithm has to be thoroughly tested. This will require a large database of

samples. However, gathering large databases of fingerprint images may be problematic due to high cost, time and privacy concern. Cappelli *et al.* [9] proposed a method to generate an artificial fingerprint image which uses the orientation map of a fingerprint as the base. Thus, this model can help to generate a large range of the artificial fingerprints through the realistically generated orientation map.

1.2 Objectives

The objective of this thesis is to present a model-based method to estimate fingerprint orientation and apply this model-based method to perform fingerprint classification. Here, three sub-questions will be discussed.

1. How to make use of the existing and reliable orientation information to recover or at least predict the missing or noisy orientation information so that the estimated orientations are robust;
2. How to precisely compute the fingerprint orientations in both local and global region in one unified model;
3. How to employ the fingerprint orientation model to perform classification.

We can regard the fingerprint orientations as vector fields. Since phase portrait model is suitable to describe the behavior of the vector fields [10–13], we attempt to investigate the use of such model to solve the identified problems. In addition, we notice that all fingerprints have a regular flow pattern in which the orientation of the flow does not change abruptly. Such smoothness constraint in the fingerprint orientation image is helpful to develop a concise and robust orientation model.

1.3 Major Contributions of the Thesis

In this thesis, the phase portrait model is adopted to perform fingerprint processing. The major contributions of this thesis are listed below.

1. An approach to detect singular points is proposed. The approach is based on the first order phase portrait model. The most attractive advantage is that it can work even with partial input only. It also provides a robust way to detect singular points which may be partially spoiled by noise.
2. A prediction model based on the piecewise first order phase portrait is proposed. The model employs two properties of fingerprints.
 - (a) The orientations near the singular points can be reconstructed using the first order phase portrait model;
 - (b) The orientations away from singular points will not change abruptly.

The advantage of this model is that we can reconstruct the global orientation with only the orientations near the singular points. Thus if there exists a large patch of noise or missing information in the fingerprint image, we can still get a coarse estimation of the orientation. A search algorithm is designed to deal with situation where not all the singular points exist in the fingerprint image.

3. In order to precisely reconstruct the fingerprint orientation, we take the first order phase portrait model as the constraints near the singular points, then adopt the nonlinear phase portrait model to reconstruct the global orientation. The advantage of this constrained nonlinear phase portrait model is that it can accurately obtain the local and global orientation at the same time. The second advantage is that the model provides a concise form to represent the fingerprint orientation.

4. A new classification method based on the constrained nonlinear phase portrait model is presented in this thesis. The algorithm uses the first 4 order coefficients of the constrained nonlinear phase portrait model as features to perform fingerprint classification.

1.4 Organization of the Thesis

The rest of this thesis is organized as follows. Chapter 2 gives an overall literature review in the areas of fingerprint enhancement, fingerprint classification, fingerprint matching, and multi-modal fingerprint processing. Subsequently the phase portrait theory will be introduced. In chapter 3, we present a new algorithm to locate the singular point using the first order phase portrait model. Chapter 4 introduces the prediction model based on the piecewise first order phase portrait model, which is followed by the constrained nonlinear phase portrait model in chapter 5. Chapter 6 proposes a fingerprint classification algorithm based on the constrained nonlinear phase portrait model. Finally, we will conclude this thesis and suggest possible future works in chapter 7.

Chapter 2

Literature Review

2.1 Fingerprints as Biometrics

2.1.1 Overview of General Fingerprint Processing and the Influence of Orientation

There has been a long history in fingerprint recognition research especially in automated fingerprint identification systems (**AFIS**). While significant progress has been made in this research area, there are still a number of research issues that need to be addressed to improve the performance further. In general, the following areas have received considerable attentions.

1. Fingerprint image enhancement: There are many factors affecting the quality of a fingerprint image, from variations in impression and skin conditions to imperfections in the acquisition devices. According to Hong and Jain [6], approximately 10 percent of the fingerprint images acquired naturally are of poor quality. These will affect the performance of AFIS significantly.

Fingerprint image enhancement attempts to remove the noise and improve the image quality. Many methods have been proposed for fingerprint image enhancement [6, 14–18]. Hong *et al.* proposed a fingerprint enhancement algorithm based on the estimated local ridge orientation and frequency, which will be used as the parameters of the even-symmetrical Gabor filter [6]. On the other hand, Jiang [14] and the Yang [15] adopted the revised low pass filter instead of Gabor filter to avoid creating spurious ridge structure. Sherlock *et al.* used Fourier domain filters to perform fingerprint enhancement [17]. Hsieh *et al.* analyzed the fingerprint image for both the global texture and the local orientation using wavelet transform, then enhanced the fingerprint image [18]. Ito *et al.* employed reaction-diffusion mechanism to enhance the fingerprint image [19].

However, there is an interdependency between getting accurate orientation estimation and enhancement: accurate local ridge orientation information cannot be obtained without any image enhancement for poor quality images, while effective image enhancement will also depend on the accuracy of local ridge orientation information. Such interdependency is very difficult to resolve. In addition, the discontinuity of the ridge caused by the minutiae and scars or large noise patches caused by sweat or wet finger will affect the local ridge orientation as well. Therefore, it is not always possible to correctly obtain the local ridge orientation information.

We observed that the fingerprint ridges flow with a certain regularity that not only forms the flow pattern used to classify the fingerprint, but at a smaller scale, will also place a global constraint on the possible orientation of the ridge at that local region. A global model of the ridges and valleys that can be constructed from partial “valid” regions can be used to correct the errors in

- the estimated orientation images, which, in turn, will help the enhancement.
2. Fingerprint classification: in some cases an automatic fingerprint identification system requires that the input fingerprint be matched with a large number of fingerprints in the database. To reduce the search time without compromising the error rate, it is desirable to classify these fingerprints in an accurate and consistent manner such that the input fingerprint needs to be matched only with a subset of the fingerprints in the database. The existing methods [1, 20–22] can be coarsely assigned to one of the following categories: Syntactic methods, approaches based on singularities, neural network approaches and others. According to the FBI's classification scheme, the fingerprint patterns can be divided into three large general groups (arch, loop, and whorl) and eight small sub-groups (plain arch, tented arch, radial loop, ulnar loop, plain whorl, central pocket loop, double loop and accidental whorl). However, majority of the fingerprints in natural distribution belong to only two patterns, loop (65%) and whorl (30%). Most automated systems currently classify fingerprints into 5 classes – plain arch, tented arch, left loop, right loop and whorl. Their natural distribution is not even, at a ratio of 5% (for the combined plain arch and tented arch), 32.5%, 32.5% and 30% respectively. Effective and fast search in a large database is needed.
 3. Fingerprint matching: fingerprint matching has been investigated using several different strategies, such as image-based [23], ridge pattern matching of fingerprint representations [24], graph-based schemes [25] and the popular minutiae matching algorithms that are based on point pattern matching [26]. Minutia arises when there is discontinuity in the ridge. For example, a ridge can suddenly come to an end (which is called termination), or can divide into two ridges (which is called bifurcation). Minutiae-based methods are most well known and

widely used for fingerprint matching. It is accepted as a proof of identity in the courts of law in almost all countries. Generally each minutia is described as a feature vector by a number of attributes, *e.g.* location in the fingerprint image, orientation, type. Two minutiae are considered "matching" if the distance between two corresponding feature vectors is smaller than a given threshold. A crucial step of minutiae-based methods is alignment (or registration) of two fingerprints. In 1996, Ratha *et al.* [27] proposed a Hough transform-based minutiae matching approach. In 1997, Chang *et al.* [28] used Hough transform to find the principal pair (which refers to the maximum Matching Pair Support). This method can reduce the number of minutiae considered during the matching stage and therefore, limit the computational complexity of the original Hough transform. In 2000, a new method which considered both the local minutiae matching and global minutiae matching was proposed by Jiang and Yau [29]. In their method, the local structures are formed by a central minutia and its two nearest-neighbor minutiae; the best matched minutia is selected and used for registering the two fingerprints. The feature vectors of the remaining aligned pairs are then matched and a final score is computed by taking into account both the local match and the global match. Instead of the triplet structure in [29], Fan *et al.* [30] performed a geometric clustering of minutiae points. Other minutiae-based works are described in Udupa *et al.* [31], Jain *et al.* [32], Hong *et al.* [33], Luo *et al.* [34]. For minutiae based fingerprint processing, insufficient number of corresponding minutiae, missing or spurious minutiae, large distortion due to variation in force used and deformation are some of the things that hinder the performance of the AFIS system. Improving the fingerprint orientations will reduce the number of missing and spurious minutiae which will also increase the performance of fingerprint matching.

4. Multi-modal processing: multi-modal biometric processing is getting popular recently because biometric systems that use a single biometric trait have to contend with noisy data, restricted degrees of freedom, failure-to-enrol problems, spoof attacks, and higher error rates. Most of the multi-modal methods are the fusion of different identification algorithms [35] and different biometric traits [36]. In addition, we find a novel mosaicking technology [37, 38]. It combines several partial impressions from the same finger into one template. The need for lower cost sensors will in future drive the sensor to a smaller size that could not cover the entire finger. So if we adopt the smaller sensor, how to improve the performance of the AFIS is a real challenge. This is even more difficult in scenario where there are no overlapping areas among the images obtained from an array of small sensors, which raises the questions as to how to align them and to recover or at least predict the lost information of the fingerprint that arises from the finite space separating one discrete fingerprint sensor from another.

2.1.2 General Methods to Deal With the Noisy Images

The most widely used approach to estimate the local ridge orientation in noisy region is low pass filtering [12]. In this method, the assumption made is that the neighboring regions surrounding the noisy area have reliable orientation. In practice, this assumption fails when the noisy patch is large. Moreover, when performing low pass filtering, the erroneous orientations in the noisy region will make the correct neighboring orientations deviate from their true value.

The alternative way to estimate the orientation in a poor quality image is to apply a set of predefined filters [33, 39]. The filter with the best response is selected

as the desired filter. This method focuses on the local region as well, so it will fail in the regions corrupted by strong oriented noise such as scars. The method is also computationally expensive as it has to perform a large number of filtering on the image.

Another commonly used approach to deal with the poor quality image is to exclude these regions from further processing. In Hong *et al.* [6], if the neighboring regions cannot provide reliable local orientation and local frequency, the regions will be marked as unrecoverable regions. There will be no further operations on these regions.

Different from the existing methods, in this work, the global orientation pattern is used to help recover the orientations in the noisy regions. The global orientation pattern has the potential to overcome the limit of the above mentioned local orientation estimation algorithm as it is less affected by the error in the local region. Consequently it will improve the enhancement algorithm by providing the correct parameters to the orientation filter.

2.1.3 Prior Works Related to Partial Image Processing

There are some prior works that deal with partial image input. Minutiae are commonly used to represent a fingerprint image for recognition. Some researchers use the minutiae to combine several partial templates [37] or partial images [38] into a large one. The former is known as "fingerprint template synthesis" while the latter as "fingerprint mosaicking". The method aims to generate a larger representation from the smaller partial inputs typically obtained from a small sensor during the enrolment phase. Hence using the enlarged template, the algorithm can deal with the partial image input during the query phase.

In 1999, Cappelli *et al.* proposed a method to perform fingerprint classification by grouping the “homogeneous” orientation image regions using MASKS and dynamic MASKS. This could be viewed as a compact representation of the orientation image. Relational graphs were then formed as the features. A number of experiments conducted on the noisy images and partial images show that the method is robust to these factors.

In this thesis, we also deal with the partial image input problem. However, we address it at a more fundamental level of orientation reconstruction. A robust orientation estimation is crucial in almost all fingerprint processing applications. As far as we know, only a few works have been reported in the literatures and even that, most of them (such as the Zero-pole model [40]) are not able to accurately compute the orientation given only the partial images. More detailed study and comparison of the past works can be found in chapter 4, sec 4.2 of this thesis.

The challenges that AFIS faces are not limited to the above areas. However fingerprint orientation is crucial in all of them. The fingerprint orientation has similar behavior as the flow field or velocity field in dynamic systems. Phase portrait model is a powerful tool to investigate the dynamic systems, therefore we will introduce the theory of phase portrait in the next section.

2.2 Phase Portrait Model Overview

The contents of the following subsections 2.2.1, 2.2.2 and 2.2.3 come from the textbook [41].

2.2.1 Introduction

Phase portrait is part of the qualitative theory of differential equation. This section will give an overview of the phase portrait theory.

Consider the system of differential equations.

$$\dot{x} = \frac{dx}{dt} = P(x, y); \quad (2.1a)$$

$$\dot{y} = \frac{dy}{dt} = Q(x, y); \quad (2.1b)$$

The two functions $P(x, y)$ and $Q(x, y)$ specify an unknown curve in the (x, y) plane. The right hand side of Eqn. 2.1 expresses the tangent vector to the curve. The qualitative behavior of the solution is determined by how $x(t)$ and $y(t)$ behave as t increases. This geometrical representation of the qualitative behavior of $[\dot{x}, \dot{y}] = \dot{X}(x, y)$ is called its **phase portrait**. The phase portrait records only the direction of the velocity of the phase point and therefore represents the dynamics in a qualitative way.

The points $\{(x_s, y_s)\}$ which satisfy the following equations:

$$\begin{cases} P(x_s, y_s) = 0 \\ Q(x_s, y_s) = 0 \end{cases} \quad (2.2)$$

are called **singular points** (also known as fixed points or critical points in geometrical theories). The existence of singular points affects the behavior of the orientation image significantly. Thus they are important for proper understanding of the orientation image. Different singular points may exhibit different behaviors.

2.2.2 Linear Phase Portrait Model

For simplicity, we first explain the first order phase portrait model. The mathematical form of the first order phase portrait model can be expressed as in Eqn. 2.3.

$$\begin{bmatrix} dx/dt \\ dy/dt \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} = \mathbf{A}\mathbf{X} + \mathbf{B} \quad (2.3)$$

The form of the $\mathbf{A}$ matrix determines the qualitative behavior of the system and a family of curves in the phase plane while the vector $\mathbf{B}$ allows for translation of the origin. Solutions to these equations are typically exponential, and the particular form depends on initial conditions. The qualitative behavior of the trajectories are classified into one of six canonical forms according to the eigenvalues of the $\mathbf{A}$ matrix as shown in table 2.1 [40].

The first step in the classification of the qualitative behavior of phase portrait is to find the characteristic polynomial $\det(\mathbf{A} - rI)$, which is quadratic. We write it as $r^2 + pr + q$, where p and q are real numbers (assuming that the matrix $\mathbf{A}$ has real entries). The classification will then depend mainly on p and q .

1. if $p^2 - 4q > 0$, there are two distinct real eigenvalues;
 - (a) if $q > 0$, there are two eigenvalues with the same sign. This type of phase portrait is called a “node”;
 - (b) if $q < 0$, there are two eigenvalues with the different sign. This type of phase portrait is called a “saddle”;
2. if $p^2 - 4q = 0$, there is only one eigenvalue $-p/2$;
 - (a) if there are two linearly independent eigenvectors, we have straight-line

Table 2.1: Classification of different phase portraits based on the nature of eigenvalues and the associated Jordan canonical forms

Case	Jordan's Form	Type of Phase Portrait	Example of Phase portrait
real distinct eigenvalues	$\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$ $\lambda_1 \times \lambda_2 > 0$	NODE
λ_1 and λ_2	$\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$ $\lambda_1 \times \lambda_2 < 0$	SADDLE
equal eigenvalues	$\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_1 \end{bmatrix}$	STAR NODE
$\lambda_1 = \lambda_2$	$\begin{bmatrix} \lambda_1 & 1 \\ 0 & \lambda_1 \end{bmatrix}$	IMPROPER NODE
complex eigenvalues	$\begin{bmatrix} A & -B \\ B & A \end{bmatrix}$ $A = 0$	CENTER
$\lambda_1 = A + jB$ $\lambda_2 = A - jB$	$\begin{bmatrix} A & -B \\ B & A \end{bmatrix}$ $A \neq 0$	SPIRAL

trajectories in all directions. This kind of phase portrait is called a “star node”;

- (b) if there is only one linearly independent eigenvector, there is only one straight line. The other trajectories are curves, which come in from the origin tangent to the straight-line trajectory and curve around to the op-

posite direction. This type of phase portrait is called an “improper node”;

3. $p^2 - 4q < 0$, there are two complex eigenvalues;

(a) if $p \neq 0$, there are two complex eigenvalues. This kind of phase portrait is called a “spiral”;

(b) if $p = 0$, there are two pure imaginary eigenvalues. This kind of phase portrait is called a “center”;

The first order phase portrait has one singular point so it is suitable to analyze the local behaviors of singular points.

2.2.3 Nonlinear Phase Portrait Model

In order to describe the more complex orientation image which may contain more than one singular points, we should increase the order of Eqn. 2.3. The phase portrait with order n can be formulated as shown in Eqns. 2.4:

$$dx = \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} x^{i-j} y^j; \quad (2.4a)$$

$$dy = \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} x^{i-j} y^j; \quad (2.4b)$$

The nonlinear description provides a more accurate representation of the orientation image than the linear description. Also it has the ability to describe the orientation image with multiple singular points since there will be up to n^2 resolutions for both dx and dy equal to 0.

In order to acquire the coefficients of the higher order phase portrait, least square algorithm will be used along with the x , y components of the orientation images

respectively.

2.2.4 Orientation Description Using Phase Portrait Model

There have been numerous previous works in describing and analyzing orientation of general images.

Martin *et al.* adopted a linear delineation to describe the “line-like” structure [10]. Weijer *et al.* used curvilinear models to analyze a wide variety of curved orientation patterns [42] while Chong *et al.* represented the orientation pattern using B-spline functions [43]. These works have been successfully applied to solve problems in their respective related fields such as detection of curve and image compression. However they are not immediately applicable to the problem of modeling fingerprint orientation pattern as it represents a curve-like (1-dimensional) process rather than a flow-like (2-dimensional).

Zucker described orientation in terms of tangent vector fields, distinguishing between Type I (contour) and Type II (flow) processes [11]. Fingerprint local ridge orientation is an example of a type II (flow) process. Furthermore, Zucker and others have developed techniques of trace inference and curve detection that have successfully detected ridges in fingerprints. The usefulness of these works is that they apply differential geometric concepts such as tangent fields, direction fields and Poincare indices to analyze directional images.

In 1987, Kass *et al.* analyzed oriented patterns by estimating the dominant local orientations and combining these to construct a flow coordinate system [12]. It is a piecewise method to describe complex orientation pattern.

Subsequently, in 1992, Jain and Rao developed the solution of signal-to-symbol transformation in the domain of flow fields, such as oriented texture fields and velocity

vector fields [44]. They used the geometric theory of differential equations to derive a symbol set based on the visual appearance of phase portraits. This paper provided a methodology to describe flow fields both qualitatively and quantitatively. The method also provided a computational framework that began with a given flow field to derive its symbolic representation. Specifically, the attractiveness of this approach is its ability to provide a reconstruction of salient features of the original texture based on the symbolic descriptor.

The advantages of this method are:

1. It is able to fully describe the orientation patterns;
2. It is possible to automatically synthesize descriptions of complex oriented patterns;
3. It can be used to make quantitative measurements in the area of experimental fluid mechanics and flow visualization;
4. It can be used for the segmentation of oriented texture pattern.

$$dx = \Lambda_1 \left[\frac{\Delta x_1(x, y)}{2\pi r_1^2(x, y)} \right] + \Lambda_2 \left[\frac{\Delta x_2(x, y)}{2\pi r_2^2(x, y)} \right] + \cdots + \Lambda_1 \left[\frac{\Delta x_n(x, y)}{2\pi r_n^2(x, y)} \right] + \Lambda_1 \cos(\theta) \quad (2.5a)$$

$$dy = \Lambda_1 \left[\frac{\Delta y_1(x, y)}{2\pi r_1^2(x, y)} \right] + \Lambda_2 \left[\frac{\Delta y_2(x, y)}{2\pi r_2^2(x, y)} \right] + \cdots + \Lambda_1 \left[\frac{\Delta y_n(x, y)}{2\pi r_n^2(x, y)} \right] + \Lambda_1 \sin(\theta) \quad (2.5b)$$

Further work by the same authors in 1993 gave the details of the estimation algorithm and error analysis for single linear oriented pattern in images [40]. Following that, in 1994, they developed a weighted estimator which is robust to noise in the orientation pattern [45]. The robustness analysis against noise is shown in Fig. 3.6.

In 1993, Ford *et al.* used the phase portraits to model the flow-like orientation pattern [46]. Compared with Jain's methods which focused on the salient feature

description, Ford *et al.* provided a framework to describe and reconstruct global complex orientation pattern. They suggested that flows were modeled as a superposition of primitives, where their associated strengths were determined from the orientation pattern. Eqn. 2.5a and 2.5b are the mathematical expression of their proposed approach. Another attribute of Ford's discussion about phase portraits is that they provided the robustness analysis in partially occluded data. They showed that even when the data was occluded by up to 40% (as shown in Fig. 3.7), the proposed phase portrait approach was still able to approximate the orientation quite accurately.

$$dx = a_{00} + a_{10}x_1 + a_{01}y_1 + a_{20}x^2 + a_{11}xy + a_{02}y^2 + \cdots + a_{ij}x^i y^j \quad (2.6a)$$

$$dy = b_{00} + b_{10}x_1 + b_{01}y_1 + b_{20}x^2 + b_{11}xy + b_{02}y^2 + \cdots + b_{ij}x^i y^j \quad (2.6b)$$

Ford *et al.* expanded their work in 1993 [47] and 1995 [48] by proposing nonlinear phase portrait models. In these papers, they decomposed the flow field into simple component flows based on the critical point behavior. Then a Taylor's model (see Eqn. 2.6(a&b)) was assumed for the velocity components, and the model coefficients were computed by considering both local critical points and global flow field behavior. Compared with the linear phase portrait, this model's advantage is that it can provide a better approximation in the critical point regions. However, their method is aimed at orientation image with a symmetric orientation structure and only two and three order phase portraits are sufficient to ensure good performance. At such low orders, the proposed approach is not able to cater to complex pattern present in the fingerprints.

From the above, we can conclude that the phase portrait model has the following advantages:

1. It is able to describe the orientation pattern, even for the complex orientation pattern;
2. It is simple to estimate the orientation near the critical points;
3. It is robust even in noisy condition or with incomplete input data;
4. It can reconstruct the orientation pattern by several methods such as the simple way of weighting the different critical point effects or the complex way of using high-order phase portrait model;
5. It can provide a method to estimate the critical points, or for fingerprint image, singular points.

Because of these advantages, we choose the phase portrait approach to model the fingerprint orientation pattern.

Chapter 3

Singular Point Detection Based On Phase Portrait

Singular points indicate the points where the orientation vanishes. In fingerprint images, there are two kinds of singular points: core point and delta point as shown in Fig. 3.1:

Singular points play three important roles in fingerprint processing:

1. Image Registration aligns two images so that they do not have position or rotation displacement. Singular points are salient structures and are limited in number (0 for arch, 2 for loop and 4 for whorl). Therefore, they can be taken as the landmark points for registration.
2. Fingerprint images can be classified into one of the several categories according to the amount, type and relative positions of singular points. These features are usually used as the coarse level features for classification.
3. Some authors employ singular points to reconstruct the fingerprint orienta-

tion [7,8]. It considers the global orientation pattern as constraints, thus more robust to noise.

Figure 3.1: The example of singular points in the fingerprint image: core point marked as $\circ$ and delta point marked as Δ

A stable and accurate detection of the singular points is crucial for fingerprint processing. In this chapter, we will discuss the detection of singular points.

3.1 Literature Review of Singular Point Detection in Fingerprint Processing

In fingerprint processing, singular point detection algorithms can be grouped into the following three classes: Poincaré Index-based, intersection-based and filter-based.

3.1.1 Poincare Index-Based Methods

Poincaré index approach was first proposed by Kawagoe and Tojo in 1984 [49]. The Poincaré index of a point is the number of complete rotations of the orientation along a closed curve around that point. Eqn. 3.1 shows the mathematical form for computing the Poincaré Index (Pin):

$$Pin(x, y) = \lim_{\varepsilon \rightarrow 0} \frac{1}{2\pi} \int_0^{2\pi} \frac{\partial}{\partial \theta} O_f(x + \varepsilon \cos \theta, y + \varepsilon \sin \theta) d\theta \quad (3.1)$$

where O_f represents the orientation image, (x, y) is the coordinate of an arbitrary point, ε represents a small positive number.

In the discrete form:

$$Pin(x, y) = \sum_{k=0 \dots 7} \Delta\theta(\theta_k, \theta_{(k+1) \bmod 8}) \quad (3.2)$$

where $\theta_k, k = 0 \dots 7$ are the orientations of the 8 neighbors of a point (x, y) , $\Delta\theta(\theta_k, \theta_{(k+1)})$ is defined as follows:

$$\Delta\theta(\theta_k, \theta_{(k+1)}) = \begin{cases} \theta_k - \theta_{k+1} & \text{if } |\theta_k - \theta_{k+1}| \leq \pi/2 \\ \theta_k - \theta_{k+1} + \pi & \text{if } |\theta_k - \theta_{k+1}| < -\pi/2 \\ \theta_k - \theta_{k+1} - \pi & \text{if } |\theta_k - \theta_{k+1}| > \pi/2 \end{cases} \quad (3.3)$$

For a closed curve the Poincaré Index takes only one of the discrete values: 0° ,

$\pm 180^\circ$, and $\pm 360^\circ$. For the singular point of a fingerprint, the Poincaré Index is:

$$Pin(x, y) = \begin{cases} 0^\circ & \text{for a nonsingular element} \\ 180^\circ & \text{for a core point} \\ -180^\circ & \text{for a delta point} \\ 360^\circ & \text{for a whorl point} \end{cases} \quad (3.4)$$

Thus we can determine whether a point is a singular point on a closed path according to its Pin value. The smallest closed path in an image with discrete pixels is 2 by 2.

In 2002, Bazen and Gerez [50] applied Green's Theorem to compute the Poincaré index. The Green's Theorem states that a closed line-integral over a vector field can be calculated as the surface integral over the rotation of this vector field:

$$\oint_{\partial A} \omega_x dx + \omega_y dy = \iint_A rot[\omega_x \omega_y]^T dx dy = \iint_A \left(\frac{\partial \omega_y}{\partial x} - \frac{\partial \omega_x}{\partial y} \right) dx dy \quad (3.5)$$

Thus the computation of the Poincaré Index as in Eqn. 3.2 becomes:

$$Pin(x, y) = \sum (J_x \triangle x + J_y \triangle y) = \sum_A \left(\frac{\partial J_y}{\partial x} - \frac{\partial J_x}{\partial y} \right) \quad (3.6)$$

where J_x and J_y are the x , y components of orientation image.

Poincaré Index method is easy to understand and implement. However it may lead to false detection in noisy images. Additional preprocessing (smoothing the orientation [51]) or postprocessing (combining other information such as quality check and segmentation [50]) is required to reduce the false detection. Another limitation of Poincaré Index approach is the computation of Pin requires a closed path. If there is no complete orientation pattern around the singular point, the detection will fail.

3.1.2 Intersection-Based Methods

Some authors took the singular point as the intersection of certain lines.

Figure 3.2: An example for localization of singular points using transition lines

Rämö *et al.* proposed a method in which singular points are taken as the intersections of transition lines [52]. Transition lines are defined as the lines on which the orientation vectors change their signs.

Considering the double orientation $2 * \theta(i, j)$:

$$DX(i, j) = \cos(\theta(i, j) * 2) \quad (3.7a)$$

$$DY(i, j) = \sin(\theta(i, j) * 2) \quad (3.7b)$$

For x component DX , the transition lines consist of the points with orientation 45° or 135° and for y component DY , these lines consist of the points with orientation 0° , 90° or 180° in the double orientation images. Fig. 3.2 shows an example of the detection of singular points using transition lines.

The advantage of this method is that it is up to 10 times faster than the Poincaré Index based method due to its lower computational complexity.

Rerkrai and Areekul proposed another type of intersection point [53]. They defined some straight lines normal to the ridge lines. Then the centroid point of the intersection points of these lines is defined the “focal point”. The focal point is demonstrated experimentally to be quite stable.

However the focal point requires sufficient amount of normal lines to ensure that the focal point converges, therefore it is time-consuming. It also requires proper division of the fingerprint image so that there is only one singular point in each region. Otherwise, multiple singular points will mix up the intersections. Fig. 3.3 shows an example of the detection of the focal point.

3.1.3 Filter-Based Methods

In 2003, Nilsson and Bigun proposed a complex symmetrical filter based method to detect singular points [54].

Figure 3.3: An example of the focal point detection approach

In their work, two types of complex filters are defined: $h_1(x, y) = (x + iy)g(x, y)$ and $h_2 = (x - iy)g(x, y)$. Here $g(x, y)$ is a gaussian function defined as $g(x, y) = \exp\{-(x^2 + y^2)/(2\sigma^2)\}$. In Fig. 3.4 we show the patterns which have the local description of $z = \exp\{\pm i\varphi\}$ and their corresponding filters, here $\varphi = \arctan(y/x)$.

Figure 3.4: Patterns and corresponding complex filters: (a) pattern with the local orientation $\varphi = \arctan(y/x)$; (b) pattern with the local orientation $\varphi = -\arctan(y/x)$; (c) filter h_1 to detect the pattern in (a); (d) filter h_2 to detect the pattern in (b)

Since the core point and delta point have similar pattern as those in Fig. 3.4(a) and (b), the authors applied the filters on the orientation image of a fingerprint. The magnitudes of the core-type (or delta-type) filter responses larger than a threshold imply the presence of the core point (or delta point). The corresponding angle of the filters' responses indicates the core's and delta's orientation. In order to accurately

detect the singular points, they employed multi-scale filtering.

The filter-based method can extract the position and orientation of the singular point at the same time. There is also potential to speed up the detection process due to the separable property of the complex symmetrical filters from one 2-dimensional filter into two 1-dimensional filters.

The magnitude of complex symmetrical filter indicates how similar the query pattern is to the symmetrical pattern. Thus a complete pattern around a candidate singular point is also a prerequisite.

3.2 Singular Point Detection Based On Phase Portrait

3.2.1 Theoretical Background

The first order phase portrait model can be used to describe the linear orientation image. Its mathematical form can be expressed in Eqn. 3.8 :

$$\begin{bmatrix} dx/dt \\ dy/dt \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} = \mathbf{A}\mathbf{X} + \mathbf{B} \quad (3.8)$$

where $\mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ and $\mathbf{B} = \begin{bmatrix} e \\ f \end{bmatrix}$

In the orientation image, the singular points denote the points where the dx and dy are 0 simultaneously. This implies that the singular point $\mathbf{X}_s(x_s, y_s)$ satisfies

Eqn. 3.9:

$$\mathbf{A}\mathbf{X}_s + \mathbf{B} = 0 \quad (3.9)$$

The position of the singular point can be obtained by:

$$\mathbf{X}_s = -\mathbf{A}^{-1}\mathbf{B} \quad (3.10)$$

According to table 2.1, the type of the singular points can be decided by the eigenvalues of the corresponding matrix $\mathbf{A}$. With the matrix $\mathbf{A}$ and $\mathbf{B}$, we can get the reconstructed orientation image $\hat{\theta}$, which should satisfy the following Eqn. 3.11

$$\tan(\hat{\theta}) = \frac{\sin(\hat{\theta})}{\cos(\hat{\theta})} = \frac{cx + dy + f}{ax + by + e} \quad (3.11)$$

From the above analysis, the key problem of estimating the position of singular points is to obtain the coefficients $\mathbf{A}$ and $\mathbf{B}$. We obtain the coefficients $\mathbf{A}$ and $\mathbf{B}$ by the Shu and Jain's method [40]. Here short introduction will be given for the completeness of the content.

Consider the Eqn. 3.12:

$$\tan(\theta) = \frac{\sin(\theta)}{\cos(\theta)} = \frac{cx + dy + f}{ax + by + e} \quad (3.12)$$

Rewrite the Eqn. 3.12 as Eqn. 3.13:

$$(a \sin(\theta) - c \cos(\theta))x + (b \sin(\theta) - d \cos(\theta))y + (e \sin(\theta) - f \cos(\theta)) = 0 \quad (3.13)$$

The least square estimator is to minimize the following Eqn. 3.14:

$$D = \sum_{i=0}^N (a \sin(\theta)x_i + b \sin(\theta)y_i + e \sin(\theta) - c \cos(\theta)x_i - d \cos(\theta)y_i - f \cos(\theta))^2 \quad (3.14)$$

where N represents the total number of the orientations.

A normalization constraint $\sqrt{a^2 + b^2 + c^2 + d^2} = 1$ is added to ensure the uniqueness of the solution.

Thus the coefficients $\mathbf{A}$ and $\mathbf{B}$ should minimize this distance D with constraint as shown in Eqn. 3.15:

$$D = L_4^T X_4^T X_4 L_4 + 2L_4^T X_4^T (L_2^T X_2^T)^T + L_2^T X_2^T X_2 L_2 + \lambda(L_4^T L_4 - 1) \quad (3.15)$$

$$\text{where } L_4 = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}, L_2 = \begin{bmatrix} e \\ f \end{bmatrix}, X_2 = \begin{bmatrix} \sin(\theta) & -\cos(\theta) \\ \sin(\theta) & -\cos(\theta) \\ \vdots & \vdots \\ \sin(\theta) & -\cos(\theta) \end{bmatrix}$$

$$\text{and } X_4 = \begin{bmatrix} \sin(\theta)x_0 & \cos(\theta)y_0 & -\sin(\theta)x_0 & -\cos(\theta)y_0 \\ \sin(\theta)x_1 & \cos(\theta)y_1 & -\sin(\theta)x_1 & -\cos(\theta)y_1 \\ \vdots & \vdots & \vdots & \vdots \\ \sin(\theta)x_n & \cos(\theta)y_n & -\sin(\theta)x_n & -\cos(\theta)y_n \end{bmatrix}.$$

In order to get the solution of the above function, we differentiate D with respect

to L_4 , L_2 and λ and obtain the following equations:

$$\frac{\partial D}{\partial L_4} = 2X_4^T X_4 L_4 + 2X_4^T X_2 L_2 + 2\lambda L_4 = 0 \quad (3.16)$$

$$\frac{\partial D}{\partial L_2} = 2X_2^T X_2 L_2 + 2X_2^T X_4 L_4 = 0 \quad (3.17)$$

$$\frac{\partial D}{\partial \lambda} = L_4^T L_4 - 1 = 0 \quad (3.18)$$

which will lead to an eigenvector problem:

$$(-X_4 X_4 + X_4^T X_2 (X_2^T X_2)^{-1} (X_2^T X_4)) L_4 = \lambda L_4 \quad (3.19)$$

The best estimation of L_4 is given by the eigenvector with the smallest absolute eigenvalue while L_2 can be computed using:

$$L_2 = -(X_2^T X_2)^{-1} X_2^T X_4 L_4 \quad (3.20)$$

In an actual fingerprint image, noise usually affects the original orientation. The estimation of the vector L_4 and L_2 can be improved by a weighted least square approach. Let $\omega(i, j)$ represents the weight at (i, j) . Replace X_4 with ωX_4 and X_2 with ωX_2 , the Eqn. 3.19 can be rewritten as Eqn. 3.21:

$$(-X_4^T \omega^2 X_4 + X_4^T \omega^2 X_2 (X_2^T \omega^2 X_2)^{-1} (X_2^T \omega^2 X_4)) L_4 = \lambda L_4 \quad (3.21)$$

Thus L_4 and L_2 are estimated using weighted least square approach. Using estimated parameters L_4 and L_2 , we can reconstruct the orientation image and locate the singular point as well.

3.2.1.1 Robustness Analysis

Using weighted least square approach, the first order phase portrait has been shown to be robust under noisy data by Shu and Jain [45] and in partially occluded data by Ford *et al.* [46]. Here we will restate it using two examples.

Case 1: Noisy Data

In the noisy case, we add random zero-mean Gaussian noise with increasing variance from 0 radian to 1.5 radian with a step of 0.1 radian. Twenty experiments were performed for each variance level. Fig. 3.5(a,b,c and d) show the average estimated parameters and the positions of the estimated singular points. When the variance of Gaussian noise is smaller than 1 radian, the estimated parameters of the first order phase portrait are accurate and stable which can be observed from small changes of mean values and small changes of variance values for increasing noise variance in Fig. 3.5.

Fig. 3.6 shows an example for the noisy case. In this example, we add gaussian noise with 0 mean and 1 radian variance and 1.5 radian variance to the synthetic data respectively. The reconstructed orientation from the noisy data is shown in Fig. 3.6(c) and (e).

The coefficients used in this example are $\{a = 1; b = 0; c = 0; d = 1; e = -15; f = -15\}$ and the estimated ones are $\{a = 0.4457; b = -0.0124; c = -0.0146; d = 0.8950; e = -7.3259; f = -14.3213\}$ for the noise of variance 1.0 radian and $\{a = 0.5601; b = -0.3494; c = -0.5256; d = 0.5366; e = -3.4667; f = -.3139\}$. The center at the synthetic orientation image is $(x_s = 16, y_s = 16)$ and at the estimated orientation image is $(\hat{x}_s = 16.8876, \hat{y}_s = 16.2771)$ and $(\hat{x}_s = 16.8450, \hat{y}_s = 17.0948)$ for the noise of variance 1.0 radian and 1.5 radian respectively.

(a) Means of the coefficients a, b, c, d (b) Variances of the coefficients a, b, c, d (c) Means of the singular point position (x_s, y_s) (d) Variances of the singular point position (x_s, y_s)

Figure 3.5: The parameters estimated from orientation images with Gaussian noise of variance from 0 radian to 1.5 radian

Case 2: Partially Occluded Data

In some cases, the orientation image includes not only the noisy data but also occluded data. In order to investigate this situation, we extend the experiments in Case 1 which can be viewed as the case of 0% occluded orientation image. In this part, we repeat the experiments in case 1 using the orientation images with various occluded portion as the input data. Fig. 3.8 show the results obtained. Here we adopt portion of the occluded orientation from 0% to 100% and variance of noise

Figure 3.6: Robustness analysis of phase portrait with noise input

from 0 radian to 0.3 radian.

Fig. 3.7(a-h) show an example where Gaussian noise with $\sigma = 0.1$ radian was added to the various occluded orientation images.

When the noise increases, the reliability of the reconstructed orientation decreases as the occluded portion increases. In general, for the low noise level ($\sigma \leq 0.2$ radian), the weighted least square approach can give a reasonable estimation of the first order phase portrait with up to 40% occluded data. Beyond this value, the estimation will become unreliable.

These results imply that good phase portrait estimation is possible for the cases where the occlusion is lower than 50% with low noise level ($\sigma \leq 0.2$ radian) and for cases where there is no occlusion but the noise level is moderately high ($\sigma \leq 1$ radian).

Figure 3.7: Robustness analysis of phase portrait with occluded data

Figure 3.8: Distance from the estimated to true singular point using occluded orientation images with Gaussian noise of variance from 0 radian to 0.3 radian

3.2.2 Prior Works on General Singular Point Detection Using Phase Portrait

We found two prior works reported in the literature which used phase portrait based approach to detect singular points. These were applied to the general oriented texture.

3.2.2.1 Isotangent-Based Algorithm

The isotangent-lines based algorithm was proposed by Shu *et al.* [55] in 1995. In their method, an isotangent-line was defined as the set of image points which shared the same tangent value of the corresponding orientations.

Let θ represents the orientation at point (x, y) in the image. Eqn. 3.8 becomes:

$$\tan(\theta) = \frac{ax + by + e}{cx + dy + f} \quad (3.22)$$

Rewrite the above equation as Eqn. 3.23

$$(a - c \tan \theta)x + (b - d \tan \theta)y + (e - f \tan \theta) = 0 \quad (3.23)$$

For any given θ , Eqn. 3.23 represents a straight line. The singular point is defined as the point with the shortest distance to all possible isotangent lines obtained when θ is varied.

In this method, an important step was to find the isotangent lines. In order to get the set of the points which share the same tangent value of the orientations, the orientations must be quantized. The method cannot be readily applied to noisy images as the noise will cause error to the isotangent lines. Another drawback similar to the "focal point" method as stated in sec. 3.1.2 is that the method requires pre-segmentation of the image into regions, each with at most one singular point. This segmentation in itself requires some rough estimation of the singular points.

3.2.2.2 Direct Estimation

The method proposed by Rao *et al.* in 1992 [44] directly estimated the singular points using Eqn. 3.10. In their method, a steepest descent search method was used to estimate the coefficients $\{a, b, c, d, e, f\}$. From the coefficients $\{a, b, c, d, e, f\}$, the direct estimation of the position of the singular point was made using Eqn. 3.10.

A measure for closeness of fit was defined between the reconstructed orientations θ_R and the original orientations θ_O as:

$$C_w = 1 - \frac{|\sin(\theta_R - \theta_O)|}{N * M} \quad (3.24)$$

where $N * M$ is the size of window over which the fitting was performed.

The type of the singular point was indicated by analyzing the coefficient A . Then for a particular singular point type t at the singular point F , the corresponding measure $V_{F,t}$ presenting the likelihood of that singular point was updated according to the following rule:

$$V_{F,t} = V_{F,t} + C_w \quad (3.25)$$

The decision whether a point was a singular point was based on the predefined threshold. When $V_{F,t}$ is larger than the predefined threshold, it was decided as a singular point. The authors also discussed possible ways to improve the singular point detection such as using the centroid of clusters with high likelihood in a singular point likelihood map. Unfortunately, there is no experiment conducted to support the proposed method.

3.3 General Procedure of the Proposed Method

The last section includes the theoretical background of using the first order phase portrait to detect the singular points and estimating the key parameters of the phase portrait. The computation of the parameters of the first order phase portrait requires only one singular point present in the orientation image.

In this section, we propose a piecewise voting algorithm based on phase portrait approach to detect the singular points in a fingerprint image. The proposed method has some similarity with Rao's method [44] but with the following crucial differences:

1. The first order phase portrait model can only be applied to the double orientation images, but not directly to the original fingerprint orientation images. (Fig. 3.9 shows an example of the double orientation image versus the original image).
2. The method proposed in this thesis adopts multi-scale processing of the orientation to verify the singular points.
3. The proposed method adopts a different measure to generate the singular point likelihood map. In Rao's method, $V_{F,t}$ was updated using the combination of the errors between the original orientations and the reconstructed orientations and the number of voting points around the singular points. However, a valid singular point that is corrupted by noise will get less voting points than a clear singular point. Similarly for a singular point at the boundary. Thus the $V_{F,t}$ for these cases would be much lower than a clear singular point, making it difficult to set one predefined threshold for all the singular points. We proposed to overcome this problem by separating these two measures into two parts. One is called "candidate map", where the value in each point in the candidate map

indicates the frequency the point is voted as the singular points. The other is called "error measure", which indicates the difference between the original orientations and the reconstructed ones.

The overall general procedure of our proposed approach is as follows.

Step 1. Computation of the original orientation

We first estimate the original orientation image θ using the following Eqn. 3.26 [56]:

$$\theta_{orig}(x_i, y_i) = \frac{1}{2} \arctan\left(\frac{\sum 2G_x G_y}{\sum (G_x^2 - G_y^2)}\right) + \frac{\pi}{2} \quad (3.26)$$

G_x and G_y is the gradient vectors along the x and y components respectively computed in the window region centered at (x_i, y_i) .

Step 2. Computation of the candidate singular point

The second step is to find all the possible singular points in the given fingerprint image. Noted that the first order phase portrait is only suitable to handle the linear orientation image, we separate the orientation image into pieces using a sliding window centered at each pixel. The size of sliding window is selected from tradeoff. If the size is too large, the window may contain two singular points; otherwise, the window may not contain enough data to exhibit the phase portrait pattern.

The original orientation θ around the core (or delta) point cannot represent a spiral (or saddle) behavior, while double orientation 2θ could (Fig.3.9). Therefore, from the double original orientation 2θ in a sliding window centered at each pixel we can estimate a position $SP = (x_s, y_s)$ and the type of this singular point. We build two maps with respect to core point and delta point, named "core candidate map(M_c)" and "delta candidate map(M_d)". For each estimated position $SP = (x_s, y_s)$ and its type, the corresponding map will be updated by $M_{c \text{ or } d}(x_s, y_s) = M_{c \text{ or } d}(x_s, y_s) + 1$.

Figure 3.9: Behaviors of singular points: 3.9(a) and 3.9(d) show the original orientation near the core and delta; 3.9(b) and 3.9(e) show the behaviors of the spiral and saddle type phase portrait; 3.9(c) and 3.9(f) show the double orientation of the core and delta.

The candidate maps M_c and M_d represent the possibility of each pixel to be a singular point. Around the true singular points, the candidate maps have higher density. Outside of the true singular points, the density is small.

Step 3. Computation of singular points

The candidate maps may include several false singular points due to noise. However these false singular points will usually have smaller values compared to the values at and around a true singular point. This is because the position of a singular point should be integer in a discretized orientation image while the estimated one is a real number. The second reason is that some singular points are estimated using the parts of phase portrait pattern around the true singular point, which may cause a small shift from the true singular point.

In order to find the true position from the candidate maps, we adopt a two-step approach:

1. Set a predefined threshold to remove the false singular points from the candidate maps. Furthermore, the isolated candidate points should also be removed. Thus the points in a small region around each true singular point are kept for further processing.
2. After most of the false singular points are removed in (1), clustering is done in each small connected region. The positions of singular points are decided by the weighted centroid of each cluster of the candidate points. Here, the values in the candidate maps act as the weights.

Step 4. Verification of singular points

The detected singular points may not be correct due to the large noisy patch or partial input such as at the boundary of the fingerprint image. Hence a verification step is involved in the proposed algorithm.

A real singular point should meet one or all of the following requirements:

1. Multi-scale property. Cores and deltas should have the same behavior at different resolution levels [57]. Hence, we re-compute the parameters of phase portrait at different resolution levels. The position should not deviate too much and the type should be consistent at all levels for a real singular point.
2. Orientation error, D , as defined in Eqn. 3.27 between the reconstructed orientation and the original one is usually small for a clear orientation image.

$$D = \frac{1}{N^2} \sum (\sin^2(\theta_{orig} - \hat{\theta})) \quad (3.27)$$

Based on our experiment, the value D should not exceed 0.15. This condition is especially important for the singular point at the boundary of the foreground

and background of a fingerprint image. The singular point at that region is hard to detect because it does not possess complete information to identify a singular point. If the reconstructed orientation and original orientation have a small difference, it indicates the existence of a singular point according to the robustness analysis of the first order phase portrait model.

This verification stage will help to confirm the true singular points and remove the false singular points. It is very useful when more than two singular points are detected for each type.

In summary, with these four steps, we could successfully find the true singular points. The robustness of the first order phase portrait model makes the detection of singular points more robust to noise and to the partial image.

3.4 Experiments

We use all the non-arch type images (3200 images in total) of NIST database 4 to evaluate our proposed method regardless of the image quality. The ground truth position of the singular points in each image are manually marked by the author for comparison. Since the resolution of the orientation image is 8×8 pixels, the detected singular point is considered correct if the distance between it and the true position is less than 16 pixels (2 orientation blocks). Otherwise the detected singular point is considered incorrect.

In order to evaluate our proposed method, we also conduct the same experiment using the Poincaré index [49]-based method, intersection based method [52] and the filter based method [54] respectively. The same error criteria is adopted for these three methods as well. Table 3.1 summarizes the results.

Table 3.1: Comparison of the accuracy of singular point detection

	True Core	True Delta	False Core	False Delta	Missed Core	Missed Delta
Poincaré-based method	4225	3943	2202	1955	680	542
Intersection based method	4225	3943	932	862	2074	1704
Filter based method	4225	3943	1118	2363	453	317
Our proposed method	4225	3943	1192	921	642	414

In order to evaluate the robustness of the proposed method against noise and partial input, we do not perform segmentation or enhancement before the extraction of the singular points for all the methods compared. However, low pass filtering is performed according to the requirement of the corresponding algorithms. The parameters of the low pass filter used in each method are selected to keep the total false detection and missed detection minimum.

Figure 3.10: Singular points detection using phase portrait for loop. Cores marked as $\circ$; Deltas as Δ ;

As shown in table 3.1, the Poincaré-based method and the intersection-based method have the highest false detection and missed detection of the singular points respectively. Checking the results shows that there are a number of false detections at the boundary of the noisy regions. If strong filtering is chosen to smooth the orientation images, the number of missed detections will increase. Furthermore, unlike our proposed approach, the other methods always require complete orientation patterns of the singular points for accurate detection. Thus they are not able to properly process the partial orientation input at the boundary of the image. On the contrary, our proposed method is able to use partial orientation information to reconstruct the orientation image using the first order phase portrait model. The reconstructed orientation image can then be used to compare with the original orientation. Thus it maximizes the information for comparison with the model of the singular points. This will result in the overall lowest error as shown in table 3.1.

Fig. 3.10 and 3.11 show the intermediate results of our proposed method. In Fig. 3.11, the delta point at the bottom left corner cannot be correctly detected using Poincaré Index as it contains only partial information due to its close proximity to the boundary. Nevertheless, it is still correctly located using our proposed approach.

More examples of the results obtained are illustrated in Fig. 3.12. Fig. 3.12 (a & b) show the singular points which are partially damaged by noise. The positions of the singular points detected using our proposed method contain small deviation from their original positions. Fig. 3.12 (c & d) show the examples where the proposed method fails due to noisy region close to the core point causing large number of erroneous orientation which affected the accuracy of the orientation computed. Fig. 3.12 (e & f) show the examples where the proposed method fails when the two singular points are close to each other (the core point position is wrongly detected).

Figure 3.11: Intermediate results of singular points detection using phase portrait for whorl. Cores marked as $\circ$; Deltas as Δ

Fig. 3.13 show the comparative examples with other prior approaches. In Fig. 3.13(a), the core and delta points can be correctly located using the proposed method. However, when the complex filter based method is adopted, the delta point cannot be correctly detected as shown in Fig. 3.13(b). Both the core and delta points cannot be detected using either the interaction-based method or the Poincare-based method as shown in Fig. 3.13(c).

Table 3.2 lists the average time taken to locate all the singular points in a single image. All the programmes are written in Matlab, version 7.1. The configuration of the computer used is Pentium (R) D CPU 3.0GHz with 1G RAM running window XP. From table 3.2, the proposed method has the highest computational cost due to its requirement to compute the eigenvalue and inverse of the matrix A . The fastest

Figure 3.12: Examples of singular points detection using phase portrait. (a & b) show the detected singular point with a small shift to the original ones; (c & d) show the examples in which the proposed method cannot correctly detect the singular points; (e & f) show the examples in which the proposed method cannot correctly detect the close singular points (In (e), the core is shifted from its original position; in (f) one core is missed). The Cores marked as $\circ$; Deltas as Δ

algorithm is the intersection-based method as it does not require the computation of the original orientation. The complex filter based method is also very fast as we

Figure 3.13: The comparative examples of singular points detection using phase portrait and other methods. (a) shows the detected singular point using the phase portrait model; (b) shows the detected singular point using the complex filter based method, the delta point is missed due to the partial corruption. (c) there is no singular points detected using the intersection-based method and the Poincare method. $\circ$ indicates core point; Δ indicates delta point

have separated the one 2D filter used into two 1D filters. Thus, the computation complexity is reduced.

Table 3.2: Computational cost comparison of the singular point detection algorithms

	Poincaré-based method	Intersection-based method	Filter-based method	the proposed method
time(s)	1.1657	0.3505	0.3602	3.02

3.5 Summary

In this chapter, we have proposed the use of the first order phase portrait model to detect and mathematically describe the singular points in fingerprint images. Although the computation cost is higher than the prior methods, the proposed approach has the following advantages:

1. It can detect the presence of singular points and describe it mathematically (i.e. model it). The presence of singular points is indicated by Eqn. 3.8 while

the reconstructed orientation image models the behavior of the singular points.

2. The first order phase portrait method does not require complete orientation pattern. Thus, it is able to detect singular points near to the boundary region or those partially corrupted by noise.

The most attractive advantage of the first order phase portrait, unlike the other approaches, is that it can reconstruct the orientation image. The reconstructed orientation provides a possibility to measure how similar the original orientation image is to the reconstructed one. This capability can be viewed as an iterative feedback mechanism. Combining the image quality information (here we adopt the orientation coherence as the quality information), we can know the source of error between the original orientation and the reconstructed orientation, and then iteratively improve the detection.

In summary, although the first order phase portrait method is not as efficient, it has the ability to reconstruct the orientation image. It is also robust against partial input, making it able to correctly detect singular points at the noisy and near the image boundary regions.

Chapter 4

Prediction Model Based On The First Order Phase Portrait

4.1 Introduction

Orientation image plays an important role in a fingerprint image processing system such as fingerprint classification, minutiae extraction, fingerprint enhancement etc.. There have been numerous reported works [5] related to computing the fingerprint orientation. However, most of the works only deal with the Local Ridge Orientation (LRO) and they are not always accurate due to noise. In addition, LRO can also be affected by the singular points such as core, delta or ridge endings and ridge breaks which have naturally sharp orientation changes. The most popular approach to reduce these errors is to impose a smoothness constraint operation in a small region. This is to ensure that the local orientation of each block does not deviate too much from the orientation of the neighboring blocks. The smoothness constraint only covers a local region. Thus if the area contain a large patch of noise, the smoothness

constraint will fail.

An alternative way to overcome the above problem will be to consider the global orientation pattern. The orientation pattern of fingerprints can be divided into several limited categories. For examples, FBI has divided the fingerprints into three large general groups and eight sub-groups for all the fingerprints [58]. For each category, the orientation pattern has its own unique regularities. Fig. 4.1 show the examples of FBI 8-class scheme.

Furthermore, within each category, the positions and behaviors of the singular points distinguish each individual fingerprint from the others. However, for the region away from the singular points the orientations are quite smooth. This characteristic can be taken as the global constraint. It implies that if we have the knowledge of singular points, we are able to give a coarse prediction of the global orientation patterns even for a region corrupted by a large patch of noise.

In this chapter, we propose an algorithm to describe the fingerprint orientation pattern based on the piecewise first order phase portrait model. This could not be solved by filter-based methods. In this model, the orientation images will be divided into several parts. In each part, there is one and only one singular point. We reconstruct the orientation image near each singular point. Then these orientations could be extended to regions away from the singular points according to the continuity of the orientation in that region. Therefore, the method has the capability to predict the orientation image even if the orientation images are corrupted by noise.

In the case that not all singular points appear in a fingerprint image, we propose a search algorithm used to predict the position of these singular points. In order to efficiently perform the search algorithm, we predefined 37 orientation patterns and compared them with the query image to coarsely determine the possible position of

Figure 4.1: Examples of FBI 8-class scheme

the singular points, followed by predicting the orientation.

4.2 Literature Review on Fingerprint Orientation Modeling

Several model-based methods have been proposed in the literature.

In 1993, Sherlock *et al.* presented an approach to model the fingerprint orientation topology [7]. It is notable as it used a simple zero-pole model to describe the fingerprint orientation pattern. In this method, the only information required is the position of the core points and the delta points. After resolving the ambiguities due to the multi-valued nature of the orientation modulo π , this model is able to describe the topology of the fingerprint orientation pattern. The basic mathematical description of this model is given in Eqn 4.1.

$$O_m(z) = O_0 + \frac{1}{2} \sum_{l=1}^L \arg(z - z_{c_l}) - \sum_{k=1}^K \arg(z - z_{d_k}) \quad (4.1)$$

where z_{c_l} and z_{d_k} are the core point positions and delta point positions respectively; z is the position of an arbitrary point; O_0 is an initial or background orientation; O_m is the orientation at z ; K and L are the numbers of delta points and core points respectively.

From Eqn. 4.1, we can observe that any two fingerprints with the same singular points will be modeled by the same function even though their local ridge orientation values may differ significantly. Furthermore, it cannot handle fingerprints without any singular point. Thus, the zero-pole model is not able to accurately model all possible fingerprint orientation patterns. Nevertheless, it gives a quite simple and elegant way to describe the global fingerprint orientation qualitatively using the directional field concept of differential geometry. The usefulness of the model is that it provides knowledge about the behavior of the orientation instead of providing

accurate orientation values.

In 1996, Vizcaya *et al.* revised the zero-pole model to a nonlinear orientation model [8]. Eqn. 4.2 describes this model in a mathematical form.

$$O_m(z) = O_0 + \frac{1}{2} \sum_{l=1}^L g_{c_l}(\arg(z - z_{c_l})) - \sum_{k=1}^K g_{d_k}(\arg(z - z_{d_k})) \quad (4.2)$$

The difference between Eqn. 4.1 and Eqn. 4.2 is the addition of the nonlinear function g . g is some family of nonlinear functions that preserve the singularity at the given point. In this nonlinear model, the orientation at any point is determined from the location of the singularities plus an additional correction term obtained for each orientation matrix. Essentially, g is decided by the mean-square orientation error between the actual orientation estimated from the fingerprint images and the orientation provided by the zero-pole model.

The paper provided a good approach in modeling the fingerprint orientation pattern. However, the proposed method of steepest gradient descent used to find the nonlinear function g is intrinsically computationally expensive. Another problem is that the proposed method depends heavily on singular points and the original orientation information. In actual practice, it is difficult to find all the singular points in a given fingerprint image. Also, the part of the original image could be too noisy to obtain the correct orientation information that can be used to compute the nonlinear function g . Nevertheless, it is noted that if the order of g is low (only a few parameters), finding g could be simplified as an optimization process in combination with the singular point detection.

Zhou and Gu [59] presented another complex model as shown in Eqn. 4.3 in 2004, which was also based on the zero-pole model while the high order rational function

Figure 4.2: Examples of zero-pole based model

$f(z)$ is used as the nonlinear correction instead.

$$O_m = \frac{1}{2} \arg \left[f(z) \frac{\prod_{l=1}^L (z - z_{c_l})}{\prod_{k=1}^K (z - z_{d_k})} \right] \quad (4.3)$$

In summary, the zero-pole model is simple to use and easy to understand. However, it can only provide a qualitative estimation of the orientation pattern instead of a precise solution. On the other hand, the nonlinear model is able to describe the fingerprint orientation pattern more accurately than the zero-pole model. The shortcoming is that it depends heavily on the availability of singular points and on the orientation pattern computed from the original input image. Furthermore, it is computationally expensive to estimate the nonlinear function from the orientation

pattern. Fig. 4.2 give some examples of these two models.

$$\cos(2\theta) = (1 \ x \ \cdots \ x^n) \mathbf{PR} \begin{pmatrix} 1 \\ y \\ \vdots \\ y^n \end{pmatrix} \quad (4.4a)$$

$$\sin(2\theta) = (1 \ x \ \cdots \ x^n) \mathbf{PI} \begin{pmatrix} 1 \\ y \\ \vdots \\ y^n \end{pmatrix} \quad (4.4b)$$

where

$$\mathbf{PR} = \begin{pmatrix} a_{00} & a_{01} & \cdots & a_{0n} \\ a_{10} & a_{11} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n0} & a_{n1} & \cdots & a_{nn} \end{pmatrix} \ \& \ \mathbf{PI} = \begin{pmatrix} b_{00} & b_{01} & \cdots & b_{0n} \\ b_{10} & b_{11} & \cdots & b_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n0} & b_{n1} & \cdots & b_{nn} \end{pmatrix} \quad (4.4c)$$

Gu, Zhou and Zhang [60] then proposed a combination model to reconstruct the fingerprint orientation image. The combination consists of a polynomial model as in Eqn. 4.4 and a point charge model as in Eqn. 4.5, where the polynomial model is used to approximate the global orientation image and the point charge model is used to correct the orientation images formed by the polynomial model near singular points. This combination model gives better estimation of the orientation compared

to the other previous models.

$$PC_{core} = \begin{cases} \frac{y-y_0}{r}Q - i\frac{x-x_0}{r}Q & r \leq R, \\ 0 & r > R \end{cases} \quad (4.5a)$$

$$PC_{delta} = \begin{cases} -\frac{y-y_0}{r}Q - i\frac{x-x_0}{r}Q & r \leq R, \\ 0 & r > R \end{cases} \quad (4.5b)$$

where Q represents the strength of the singular points and r denotes the radius of the region which the point charge model can affect.

4.3 Orientation Behavior Analysis of Fingerprints

(a) Definition of Pattern Area (b) Default Orientation Pattern (c) Default Global Orientation

Figure 4.3: Default global orientation behavior

4.3.1 Default Global Orientation Model

In a fingerprint image, a “pattern area” of the fingerprint impression is defined as the fingerprint region that forms part of the loop or whorl in which appears the cores, deltas, and ridges [58]. It is the pattern area of the fingerprint that provides

Figure 4.4: Default global orientation behavior with varying value of parameters a and b

differences in the orientation pattern. Outside of the pattern area, the fingerprint orientation pattern appears to be similar: The ridges enter from one side of the impression and flow or tend to flow out at the other side with a rise or wave at the center. We call such pattern as the “default global orientation pattern”. Eqn. 4.6 gives the phase portrait description of the default global orientation model. This model can be used to correct the local orientation pattern so that an overall smooth

pattern is obtained.

$$\begin{cases} dx = a * |y| \\ dy = b * x \end{cases} \quad (4.6)$$

where $a * b < 0$, a and b are constants; (x, y) are the coordinates of a point; (dx, dy) are the x and y components of an orientation image respectively. Fig. 4.4 give the examples of the global orientation patterns when the parameters a and b change.

4.3.2 Orientation Behavior Near Singular Points

In fingerprints, there are two important types of singular points: core point and delta point. For the arch class, the core point is not available. However we can find a reference point with maximum change in curvature as the core. These singular points are important in modeling the orientation patterns of fingerprints as the orientation patterns around these points show their unique characteristics.

The original orientation has a π step when it varies from $\pi/2$ to $-\pi/2$. This causes discontinuity in the orientation images. In order to remove this discontinuity, we use the double orientation image, denoted by 2θ , which expands the original orientation range from $[-\pi/2, \pi/2)$ to $[-\pi, \pi)$.

Bazen and Gerez [50] reported a reference model to describe the double orientation behaviors of the singular points as shown by Eqn. 4.7 ($a, b, c \& d$):

$$DX_{Core,ref} = \frac{-(y - y_0)}{\sqrt{(x - x_0)^2 + (y - y_0)^2}} \quad (4.7a)$$

$$DY_{Core,ref} = \frac{(x - x_0)}{\sqrt{(x - x_0)^2 + (y - y_0)^2}} \quad (4.7b)$$

$$DX_{Delta,ref} = \frac{-(y - y_0)}{\sqrt{(x - x_0)^2 + (y - y_0)^2}} \quad (4.7c)$$

$$DY_{Delta,ref} = \frac{-(x - x_0)}{\sqrt{(x - x_0)^2 + (y - y_0)^2}} \quad (4.7d)$$

where (x_0, y_0) are the coordinates of a singular point. $DX_{Core,ref}, DY_{Core,ref}$ are the x and y components of the reference core; $DX_{Delta,ref}, DY_{Delta,ref}$ are the x and y components of the reference delta.

However, the reference model describes only the constant form of the core and delta behaviors regardless of the local patterns around them. In other words, this reference model assumes that all the core (or delta) regions have the same behavior and therefore unable to model the local variation in the singular point region.

In fact, different singular points have their own distinct behaviors. In the following sections, we will introduce two methods based on the phase portrait model to describe the behaviors of the singular points.

4.3.2.1 The First Order Phase Portrait Model

In order to get the various behaviors of the singular points, we rewrite Eqn. 4.7(a-d) as Eqn. 4.8:

$$\begin{bmatrix} dx \\ dy \end{bmatrix} = \begin{bmatrix} c & d \\ a & b \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} f \\ e \end{bmatrix} = \mathbf{Ax} + \mathbf{B} \quad (4.8)$$

where dx and dy are the x , y components of the orientation image; (x, y) are the coordinates and $[a, b, c, d, e, \text{ and } f]$ are the coefficients of the phase portrait model. The eigenvalues of the matrix $\mathbf{A}$ determine the type of singular points and $\mathbf{B}$ allows the translation of the singular points. Hence the reference models of the singular

points are regarded as special cases of the first order phase portrait model. For example, when $\{c = 0; d = 1; a = -1; b = 0\}$ are satisfied, the Eqn. 4.8 represents the same behavior as Eqns. 4.7a & b and when $\{c = 0; d = -1; a = -1; b = 0\}$ are satisfied, Eqn.4.8 represents the same behavior as Eqns. 4.7c & d. Thus the double orientation image of the singular points can be modeled by the first order phase portrait. The coefficients $[a, b, c, d]$ can be obtained using the least square algorithm.

Figure 4.5: Behaviors of singular points: 4.5(a) and 4.5(f) show the original orientation near the core and delta; 4.5(b) and 4.5(g) show the behaviors of the spiral and saddle type phase portraits; 4.5(c) and 4.5(h) show the double orientation of the core and delta; 4.5(d) and 4.5(i) show the reconstructed orientation by the piecewise phase portrait; 4.5(e) and 4.5(j) show the reconstructed orientation by the 1st order phase portrait.

4.3.2.2 Piecewise Phase Portrait Model

In some fingerprint images, the core and delta points are located so close to each other that their orientation images interact. As such, the first order phase portrait model is not sufficient to accurately describe their orientation patterns. In order to solve this problem, we suggested a piecewise description to reconstruct the orientation images

near the singular points. This approach works on the original orientation image and its description can be obtained from partial information.

Though the original orientation pattern at a core point (as shown in Fig. 4.5(a)) cannot fit to any of the phase portrait type readily, we can divide the core region into two half planes – the upper and lower plane (as shown in Fig. 4.6(a)). The upper half plane of the core contains an orientation pattern that fits well the spiral type of phase portrait. The lower half plane region exhibits a predominant direction where most of the ridges will flow with such a direction. We call this predominant direction the “primary core direction”.

Using the above observations, we can formulate a methodology to describe the orientation pattern at the core region. In one half plane of core, the orientation pattern can be described by the spiral type phase portrait model and in the other half plane by its “primary core direction”.

Figure 4.6: Orientation behaviors near singular points

Similarly for the delta points as shown in Fig. 4.5(f), we cannot find a simple phase portrait model to represent their orientation pattern. However, as the name “delta” implies, we observe that we could divide the area around the delta point into three regions using three straight lines intersecting at the delta point (as shown in 4.6(c)). In each sub-region, the orientation pattern could be viewed as a part of the saddle type of phase portrait model. We call the three straight lines the “three

asymptotes". From the three asymptotes we can reconstruct the orientation pattern at the delta region.

The piecewise phase portrait model for the singular points is as follows:

For the core point, we divide the core into four quadrants Ω_k , $k = 1, 2, 3, 4$ as shown in Fig. 4.6(b).

Let $\theta_{-V_i} = Var(\theta_{\omega_i})$, where θ_{-V_i} is the variance of the orientations θ_{Ω_i} in the i^{th} quadrant, $\theta_P = (\Sigma\theta)/N$, where θ_P is the predominant orientation of the core in which most of the ridges tend to flow, θ is the orientation in the quadrant Ω_k where the corresponding θ_{-V_k} has the minimum value and N is the number of orientations in Ω_k .

Then the core region is divided into two half planes by a line passing through the core with a slope of $\tan(\theta_P + \pi/2)$. The line is shown as the white line in Fig. 4.6(a).

In the top left plane, the orientations are reconstructed using the coefficients of the first order phase portrait model. In the bottom right plane, the orientations are represented as the primary core direction.

For the delta point, we divide the delta region into three sub-regions using three prefixed lines with the slope of 30° , 150° and 270° counterclockwise from the horizontal and emanating from the delta point as shown in Fig. 4.6(d). In each sub-region, let $\theta_{asympt} = (\sum_{\Omega} \theta)/N$, where Ω indicates the sub-region in which the average orientation is computed; θ is the orientation of a pixel contained in sub-region Ω , N is the number of orientations in sub-region Ω ; and θ_{asympt} represents an orientation of a asymptote. Three such asymptotes are shown in Fig. 4.6(c).

Hence three asymptotes are used to divide the delta region into three parts. In each part, the orientation pattern can be viewed as a part of the saddle type of phase portrait model. From any two asymptotes we can reconstruct the orientation pattern

at the delta sub-region according to the following Eqn. 4.9 (detail of derivation is given in the appendix A):

$$\frac{dy}{dx} = -\frac{(K \cos^2 \theta - \sin^2 \theta)(x - x_0) + (K \cos \theta \sin \theta + \cos \theta \sin \theta)(y - y_0)}{(K \cos \theta \sin \theta + \sin \theta \cos \theta)(x - x_0) + (K \sin^2 \theta - \cos^2 \theta)(y - y_0)} \quad (4.9)$$

where $K = \tan^2(\frac{\theta_1 - \theta_2}{2})$, $\theta = (\theta_1 + \theta_2)/2$ and $(y - y_0) - \tan \theta_1(x - x_0) = 0$ and $(y - y_0) - \tan \theta_2(x - x_0) = 0$ represent two asymptotes, (x_0, y_0) represents the position of delta. Repeating the above procedures for the remaining two sub-regions of the delta, we are able to reconstruct the orientation image at the delta point.

Among the three methods outlined, reconstruction of the orientation pattern using the reference model is the simplest. However it cannot describe the different orientation behavior in detail for the same type of the singular points. The phase portrait model is the most robust method even for partial input and noisy input. As for the piecewise phase portrait model, it is as robust as the phase portrait model. In addition, it can also handle the situation where there exist two singular points close to each other and is thus used to implement our prediction model.

4.4 Prediction Model Applied to the Complete Input

In this section, we propose a prediction model for fingerprint orientation. The first order phase portrait model is used to describe the behavior of its singular points, which is then combined in a piecewise manner with the default global pattern to represent the global fingerprint orientation. The parameters of this model can be obtained using just the regions near the singular points. Complementing it with the regularity of the fingerprint orientation patterns, we are able to reconstruct the

orientation beyond the known region. Therefore this model has prediction capability. The basic assumption made here is that all singular points appear in the acquired image and can be located.

4.4.1 Criteria for Piecewise Model

The fingerprint orientation pattern cannot be completely described using only one first order phase portrait due to the existence of multiple singular points. Instead, we describe the model of the fingerprint orientation image in a piecewise manner.

The following rules are formulated to effectively form the whole model of a fingerprint orientation:

1. If there is no delta point in the fingerprint impression, the image will be divided into 2 sections by a horizontal line through its reference point. Then in each section, we can view its orientation pattern as one type of phase portrait model and reconstruct the orientation by its corresponding first order phase portrait. Fig. 4.7(a) illustrates this case; However, if there is no singular point, the reference point can be a point with maximum curvature. This can be viewed as a part of the first order phase portrait where the singular point is located outside the image. Based on the robust analysis of the occluded input (see Fig. 3.7), the orientation estimated will not be as accurate as when the orientation is reconstructed using a valid core point. The less abrupt the orientation is at the reference region, the less reliable is the estimated orientation.
2. If there is one delta point in the fingerprint impression, the range influenced by the core and delta is divided into two sub-regions as shown by the white line in Fig. 4.7(b) and 4.7(c). In each subregion, the orientations will be reconstructed by the corresponding singular points as stated in section 4.3.

3. If there are two delta points and two core points, the asymptotes are used to partition the lower region as shown in Fig. 4.7(d). Then in each part, the orientations will be reconstructed by the behaviors of its nearest singular points.

(a) Predicted Orientations of Plain Arch

(b) Predicted Orientations of Loop

(c) Predicted Orientations of Tented Arch

(d) Predicted Orientations of Plain Whorl

Figure 4.7: Examples of the piecewise prediction model; the behaviors of singular points can be delivered by the orientations in the white rectangle; beyond white rectangle region, the orientation can be predicted.

4.4.2 Rectification of Prediction Model

Till now, we have obtained prediction model to describe the fingerprint orientation pattern. However, there exist small regions where the orientations are not described accurately due to the piecewise consideration of the proposed model. Thus the whole orientation model needs to be rectified to improve the model further so that it will be able to match the actual orientation pattern closely.

The improvement includes two aspects:

1. Discontinuity removal: because our proposed method is piecewise, the orientation at the boundary of each piece region has to be smoothed to remove the discontinuity.
2. Weighted piecewise model: we note that there exists consistent error in the region away from singular points. Therefore, we add the default global orientation model and weight it with the local orientation model to rectify the local orientation description.

4.4.2.1 Discontinuity Removal

Discontinuity can usually be effectively removed through a smoothing operation. Since the local ridge orientation varies slowly in a local neighborhood without any singular point, a low pass filter can be used as the smoothing operator. In order to perform the low-pass filtering, the orientation model has to be converted into a continuous vector fields as follows:

$$\begin{cases} dx(i, j) = \cos(2\theta(i, j)) \\ dy(i, j) = \sin(2\theta(i, j)) \end{cases} \quad (4.10)$$

where (dx, dy) are the x and y components of the vector fields respectively. From the resulting vector fields, low-pass filtering can then be performed. Here we adopt average filter as the preferred low pass filter.

$$\begin{cases} \hat{dx}(i, j) = 1/w_\phi^2 \sum_{u=-w_\phi/2}^{w_\phi/2} \sum_{v=-w_\phi/2}^{w_\phi/2} dx(i-u, j-v) \\ \hat{dy}(i, j) = 1/w_\phi^2 \sum_{u=-w_\phi/2}^{w_\phi/2} \sum_{v=-w_\phi/2}^{w_\phi/2} dy(i-u, j-v) \end{cases} \quad (4.11)$$

where w_ϕ is the size of window in which the filtering is performed.

Thus the computation of the corrected orientation model is as shown in Eqn. 4.12:

$$\theta_{smooth}(i, j) = \frac{1}{2} \tan^{-1} \left(\frac{\hat{dy}(i, j)}{\hat{dx}(i, j)} \right) \quad (4.12)$$

4.4.2.2 Weighted Piecewise Models

Although we have constructed a model of the singular point, we note that there exists consistent error in the region away from the singular points. The error at the boundary of each sub-region can be corrected through the discontinuity removal process described in the earlier section.

From Fig. 4.7 we observe that the reconstructed orientations by the piece wise first order phase portrait model tend to be encircled at the singular point region. The predicted orientations at the edge of orientation also have the same focus. On the contrary, the orientations at the edge of the real image tend to be flat as shown in Fig. 4.3(b). Hence, such discrepancy cannot be corrected using the above method. We note that the default global model mentioned in Sec. 4.3.1 tends to be flat at the edge of the orientation image. Thus, we can assign a small weight to the default

global orientation model and a large weight to the piecewise model near the singular point. The summation of these two orientations will overcome this problem.

Since the local orientations are constructed in a piecewise manner and each singular point may have different effect on the final model, the rectified model will be piecewise and different weights will be assigned to the default global model and the piece wise phase portrait model. The section below explains the rectification proposed for the various fingerprint classes.

Rectification of the arch class models

The orientation pattern of the arch type is usually symmetric or almost symmetric at the vertical line passing through the reference point. Therefore, we have proposed the weights according to the Eqn. 4.13 which will rectify the orientation so that the model is also symmetrical.

$$\begin{cases} W_g(x, y) = \left(\frac{x - x_r}{x_{max}} \right)^2 \\ W_l(x, y) = 1 - W_g \end{cases} \quad (4.13)$$

where $x_{max} = \max|x - x_r|$, x_r is the horizontal ordinate of the reference point, $W_g(x, y)$ and $W_l(x, y)$ are the default global orientation weights and local model weights respectively.

Then the final orientation will be decided by the following Eqn. 4.14:

$$\theta(x, y) = W_g(x, y) * \theta_g(x, y) * K1 + W_l(x, y) * \theta_l(x, y) * K2 \quad (4.14)$$

where $\theta_g(x, y)$ and $\theta_l(x, y)$ are the orientations from the global model and local model respectively. $K1$ and $K2$ are factors that fit the model to the actual orientation pattern obtained from an image and can be estimated from the weighted default

global model data and weighted local model data with the data computed from the input image using the LMS algorithm. Fig. 4.8 shows the weighted orientation model of the default global and local models.

Figure 4.8: Weighted models for the arch class

Rectification of the loop class models

The loop class contains both the core and delta points. The default global model produces different effects at the core point area and the delta point area. Thus, the weights will again be piecewise. In the core point sub-region, the weights will be decided by the following Eqn. 4.15:

$$\begin{cases} W_g(x, y) = \left(\frac{x - x_{core}}{x_{max}} \right)^2 \\ W_l(x, y) = 1 - W_g \end{cases} \quad (4.15)$$

where $x_{max} = \max(|x_i - x_{core}|)$ and $W_g(x, y)$, $W_l(x, y)$ are the default global orientation weights and local model weights respectively. Then the rectified orientations in this sub-region will be decided by Eqn. 4.14.

In the delta point sub-region, the weights will be decided by the following Eqn. 4.16:

$$\begin{cases} W_g(x, y) = \left(\frac{x - x_{delta}}{x_{max}} \right)^2 \\ W_l(x, y) = 1 - W_g \end{cases} \quad (4.16)$$

where $x_{max} = \max(|x_i - x_{delta}|)$, and $W_g(x, y)$ and $W_l(x, y)$ are the default global orientation weights and local model weights respectively. The rectified orientations in this sub-region will be decided by Eqn. 4.14.

Figure 4.9: Weighted models near the delta point of the loop

Fig. 4.9 shows the weighted orientation model of the default global and local models.

Rectification of the whorl class models

For the whorl type, the orientation in each sub-region will be rectified by both the weighted default global orientation and the weighted local orientation. The weights in each sub-region will be decided by its nearest singular point, using Eqns. 4.15 and 4.16.

Then similar approach is used to rectify the orientation as given by Eqn. 4.14.

The final weighted model will be the summation of these sub-regions.

Some sample results of the rectification are shown in Fig. 4.10. From the sample results, given all the positions of singular points in the fingerprint impression, the model of the orientation can be accurately reconstructed using the proposed weighted piecewise phase portrait approach. However we notice that for most of the actual fingerprint impressions available, majority of them do not contain all singular points. Nevertheless, the results are promising as they show that the proposed approach, although simple and fast, are already able to model the general classes found in the

Figure 4.10: Examples before and after the rectification

Figure 4.10: Examples before and after the rectification(Con'd)

fingerprint impressions. The proposed approach could also be extended further to include other odd orientation patterns that are not commonly encountered. The next section will introduce an approach for the cases when not all singular points are available.

4.5 Prediction Model Applied to the Partial Input

In section 4.4, we have introduced a piecewise model based on the first order phase portrait model. The proposed model assumes that all the singular points in a fingerprint impression are available. However, this assumption is not usually valid for most of the impressions obtained using a commercial sensor due to its limited size. This section will introduce a new method to estimate the fingerprint orientation pattern when not all the singular points in the fingerprint impression are available. The basic assumption made is that we are able to find a reference point in the fingerprint impression.

4.5.1 Search Model for the Incomplete Image

In this section, we propose a method to deal with the case where not all the singular points are present in the image. The basic idea is that we first assume a singular point in the orientation image and obtain a coarse orientation pattern by the method in section 4.4. Subsequently we compute the distance between the real orientations and the coarse orientations. For all the possible singular points, the assumed points with the minimum distance will be the correct singular points.

4.5.1.1 Distance Measure

A basic requirement for the search algorithm is the distance measure between two orientation images. Since the orientation estimation algorithm gives rise to an orientation image which is not a vector field, the orientation estimated is inherently ambiguous because the texture flow at any given point could be either in the direction θ or $\pi + \theta$. (Details can be found in section II-F, [44]). Fig 4.11 shows some

possible cases of ambiguity in the orientation information.

Figure 4.11: Ambiguity of orientation image

A suitable distance measure for the search algorithm should be independent of these sign ambiguities. Therefore, we define the orientation distance measure as follows $Dist_\theta$:

$$Dist_\theta = \sum_i \sum_j \sin^2(\theta_M(i, j) - \theta_I(i, j)) \quad (4.17)$$

where θ_M is the orientation computed from the model; θ_I is the orientation estimated from the original impression.

4.5.1.2 Searching the Best Fit Model

a) Plain arch models

Using the piecewise orientation model, the orientation pattern can be determined from the position of the reference point. Thus once the position of the reference point is known, we can reconstruct the orientation image using the approach proposed in section 4.4.

b) Loop and tented arch models

Up to now, we are able to reconstruct the orientation pattern with both the core point and delta point available. However, if the delta point is not available

in a fingerprint impression, its position has to be estimated. A search algorithm is developed to solve such a problem.

In the reconstruction algorithm (see section 4.4) proposed for the orientation pattern of the loop or tented arch, 5 parameters are required at the region of the delta point. These are the x and y positions of the delta point (X_{delta}, Y_{delta}) and three asymptotes: $Asymp_{up}$, $Asymp_{left}$, $Asymp_{right}$. The best-fit model can then be obtained by minimizing the distance measure $Dist_{\theta}$ of these 5 parameters.

The following is the description of the pseudo-code for the proposed search algorithm:

Table 4.1: Pseudo-code for search algorithm

```

{
  Initialize  $X_{delta}, Y_{delta}, Asymp_{up}, Asymp_{left}, Asymp_{right}$ ;
   $Flag = 1$ ;
  While { $Flag$ }
  {
    Initialize  $Dist_{\theta} = 1$ ;
    for  $i = 1 : x_{max}$  (the maximum coordinate in x direction)
       $\theta_{M,i} = Model(i, Y_{delta}, Asymp_{up}, Asymp_{left}, Asymp_{right})$ 
       $Dist_{\theta_i} = \sum_{m=1}^{x_{max}} \sum_{n=1}^{y_{max}} \sin^2(\theta_{M,i}(m, n) - \theta_i(m, n)) / (x_{max} * y_{max})$ 
    end for
    find  $X_{delta\_temp}$  with  $Dist_{\theta_{X_{delta\_temp}}} = \min(Dist_{\theta_i})$ 

    Initialize  $Dist_{\theta} = 1$ ;
    for  $i = 1 : y_{max}$  (the maximum coordinate in y direction)
       $\theta_{M,i} = Model(X_{delta\_temp}, i, Asymp_{up}, Asymp_{left}, Asymp_{right})$ 

```

$$Dist_ \theta_i = \sum_{m=1}^{x_{max}} \sum_{n=1}^{y_{max}} \sin^2(\theta_{M,i}(m, n) - \theta_i(m, n)) / (x_{max} * y_{max})$$

end for

find Y_{delta_temp} with $Dist_ \theta_{Y_{delta_temp}} = \min(Dist_ \theta_i)$

Initialize $Dist_ \theta = 1;$

for $\theta_{up} = \theta_{up,min} : \theta_{up,max}$

$$\theta_{M,\theta} = Model(X_{delta_temp}, Y_{delta_temp}, \theta_{up}, Asymp_{left}, Asymp_{right})$$

$$Dist_ \theta_{\theta_{up}} = \sum_{m=1}^{x_{max}} \sum_{n=1}^{y_{max}} \sin^2(\theta_{M,i}(m, n) - \theta_i(m, n)) / (x_{max} * y_{max})$$

end for

find $Asymp_{up_temp}$ with $Dist_ \theta_{Asymp_{\theta_{up_temp}}} = \min(Dist_ \theta_{\theta_{up}})$

Initialize $Dist_ \theta = 1;$

for $\theta_{left} = \theta_{left,min} : \theta_{left,max}$

$$\theta_{M,\theta} = Model(X_{delta_temp}, Y_{delta_temp}, Asymp_{up_temp}, \theta_{left}, Asymp_{right})$$

$$Dist_ \theta_{\theta_{left}} = \sum_{m=1}^{x_{max}} \sum_{n=1}^{y_{max}} \sin^2(\theta_{M,i}(m, n) - \theta_i(m, n)) / (x_{max} * y_{max})$$

end for

find $Asymp_{left_temp}$ with $Dist_ \theta_{Asymp_{left_temp}} = \min(Dist_ \theta_{\theta_{left}})$

Initialize $Dist_ \theta = 1;$

for $\theta_{right} = \theta_{right,min} : \theta_{right,max}$

$$\theta_{M,\theta} = Model(X_{delta_temp}, Y_{delta_temp}, Asymp_{up_temp}, Asymp_{left_temp}, \theta_{right})$$

$$Dist_ \theta_{\theta_{right}} = \sum_{m=1}^{x_{max}} \sum_{n=1}^{y_{max}} \sin^2(\theta_{M,i}(m, n) - \theta_i(m, n)) / (x_{max} * y_{max})$$

end for

find $Asymp_{right_temp}$ with $Dist_ \theta_{Asymp_{\theta_{right_temp}}} = \min(Dist_ \theta_{\theta_{right}})$

if $([X_{delta_temp}, Y_{delta_temp}, Asymp_{up_temp}, Asymp_{left_temp}, Asymp_{right_temp}])$

```

== [Xdelta, Ydelta, Asympup, Asympleft, Asympright])
    {Flag = 0;}
else
    {Asympup = Asympup_temp; Asympleft = Asympleft_temp;
    Asympright = Asympright_temp; Xdelta = Xdelta_temp; Ydelta = Xdelta_temp; }
Endif
}End while
}

```

where $Model(X_{delta}, Y_{delta}, Asymp_{up}, Asymp_{left}, Asymp_{right})$ is a function used to reconstruct the orientation image from the model. θ_i is the orientation image computed directly from the impression and $[X_{delta_temp}, Y_{delta_temp}, Asymp_{up_temp}, Asymp_{left_temp}, Asymp_{right_temp}]$ are the temporary variables; $[X_{delta}, Y_{delta}, Asymp_{up}, Asymp_{left}, Asymp_{right}]$ are the final estimated parameters.

The search algorithm first assumes an initial position for the five parameters $[X_{delta_temp}, Y_{delta_temp}, Asymp_{up_temp}, Asymp_{left_temp}, Asymp_{right_temp}]$; The search is done by letting one of the five parameters taking all the possible values, keeping the other parameters unchanged. Here X_{delta_temp} is used as an example and the search is done over 1 to x_{max} , the maximum coordinate in x direction. Subsequently, we compute the reconstructed orientations $\theta_{M,i}$ using the prediction model ($\theta_{M,i} = Model(i, Y_{delta}, Asymp_{up}, Asymp_{left}, Asymp_{right})$). The value which minimize the distance $Dist_{\theta}$ between the reconstructed orientations and the original orientations will be taken as the updated one. Then repeat the above steps for each individual parameter by varying that parameter only and keeping all the others invariant, with the updated value minimizing $Dist_{\theta}$. Finally, repeat the above steps until all the five parameters do not change, whereby the model has the minimum distance from

the original orientation. Using these five well-tuned parameters, we can find the best-fit model for the corresponding orientation pattern.

c) Whorl models

For the whorl type of fingerprint impression, there exist at least two delta points. Since the proposed orientation model is piecewise in accordance with the number of singular points, the search algorithm could be implemented piecewise as well. Once the reference point is found, the whole impression will be divided into 3 sections as shown by section *A*, *B* and *C* in the example Fig. 4.12. The search for delta points will then be performed in sections *B* and *C*.

Figure 4.12: Division of the whorl fingerprint type into sections to facilitate the search algorithm to detect the missing delta points

Once the estimated locations of delta points have been found, the correction model described in section 4.4 can be used to refine the orientation image.

4.5.1.3 Speeding Up the Search Algorithm

The search algorithm assumes that the type of fingerprint is known. If it is not true, the search algorithm will probe for all possible positions of singular points, thus increasing the cost of computation. In order to speed up the search algorithm,

Figure 4.13: Possible positions of singular points in predefined models

we divide the pattern area into 12 blocks as shown in Fig. 4.13 based on trial and error. The center of each block is taken as the potential singular points. With these potential singular point positions, we define a total of 37 models with 1 model for plain arch, 6 models for left loop, 6 models for right loop and 24 models for whorl. There is only one model for arch type as there is only one reference point. The delta point position is ignored. For the left loop, there is one core point and 6 possible locations of the delta point located to the right of the core point as shown in Fig. 4.13, giving a total of 6 models. Similarly for the right loop with 6 possible locations to the left of the core point. For whorl, there are two core points and two delta points. Each core point will have all the 12 possible positions of the delta points as shown in Fig. 4.13. Thus, we have a total of 24 models for whorl when two core points are considered. The overall total number of models is thus 37. Using these models, we can compute the corresponding distance between the orientations estimated from the impression and the orientations computed from the model. Then the position of potential singular points with the minimum distance will be taken as the true singular points.

Then the search algorithm is only performed on the block with the minimum distance. The average complexity of the computation will be reduced to 1/12 of the

original one.

4.6 Experiments

4.6.1 Prediction Model Applied to the Complete Input

In this section, we test our prediction model on the complete input where we can locate all the singular points. Two types of experiment are conducted; one is to predict the orientation at the peripheral region and the other is to predict the orientations at the pattern area.

In order to evaluate the effectiveness of the proposed orientation prediction algorithm quantitatively, we compute the original orientation image using a Gabor filter bank (64 filters). Then the mean absolute error (MAE) between the predicted orientation and the original orientation is computed as follows:

$$MAE = \frac{1}{N} \sum_{(x,y) \in \Omega} d(\theta_{pred}(x, y) - \theta_{gabor}(x, y)) \quad (4.18)$$

where Ω is the region of comparison, which contains a total of N points. (x, y) is the coordinate of a point in Ω , θ_{pred} and θ_{gabor} denote the predicted orientation using the proposed algorithm and the orientation image computed using the Gabor filter bank respectively. Since the orientation is in $[-\pi/2, \pi/2]$, the function $d(\cdot)$ is defined as

$$d(\theta) = \begin{cases} |\theta|, & |\theta| < \pi/2 \\ \pi - |\theta|, & \text{otherwise.} \end{cases} \quad (4.19)$$

The experiments are conducted on the first 500 images (f0001-f0250 and s0001 - s0250) of the NIST special Database 4 where the fingerprint image size is relatively

large (512 by 480 pixels). Here the orientations are computed using an 8 by 8 pixel block. In experiment I and II, the data used to predict the orientation is obtained using a 9 by 9 window centered at the singular point. The size of the complete orientation is 64 by 60.

Experiment I: Peripheral orientation prediction

In the first experiment, we designate the peripheral region as the region within 128 pixels from the boundary region of the fingerprint. The orientations at the peripheral region are then removed from the images. Subsequently, the orientation in this region is predicted using our proposed approach. The average *MAE* obtained is 21.43° and its standard deviation is 9.05° before rectification. After the predicted orientations are rectified, the results are improved to 18.78° with standard deviation of 8.56° . There is improvement in the accuracy of the predicted orientation as the rectification process aims to eliminate the discontinuities between the different pieces and to make the boundary orientations close to the global orientation. The main reason for the error is that the first order phase portrait has limited description capability. These errors are shown at the bottom-left side and middle-right side in Fig 4.14(c).

Experiment II: Orientation prediction at pattern area

The second experiment is conducted using the same dataset as in Experiment I. However, the orientations with a size 10 by 10 blocks near the pattern area are removed instead of the peripheral orientation as in the first experiment. The average *MAE* obtained is 14.2458° and its standard deviation is 5.7406° before rectification. After rectification of the prediction orientations, the *MAE* obtained is 14.2043° with a standard deviation of 5.0465° . There is no obvious improvement observed. This is because the orientations at the pattern area are close to the singular points. The weights around the singular points are near to 1. Thus the rectification process give

Figure 4.14: Examples of prediction algorithm. The predicted orientation lies outside the rectangle. The images (a to d) show the coarse predicted orientation using phase portrait. (a) belongs to plain arch; (b) belongs to tented arch; (c) belongs to loop and (d) belongs to whorl.

equal values to the original and predicted orientations. The main reason why the MAE is not close to zero is that the first-order phase portrait has limited capability to describe the oriented pattern. Moreover, the predicted orientation at the pattern area are more accurate than at the peripheral region, with lower MAE . Thus the improvement is limited. Some of the examples are shown in Fig. 4.15.

Figure 4.15: Sample performance of the prediction algorithm. The predicted orientation lies inside the dark area. The images in the last column show the coarse predicted orientation using phase portrait. The images in each row are the same. Images in the first row belong to the plain arch; the second row to tented arch and the last row to whorl.

The overall MAE and its standard deviation are listed in the table. 4.2. More comparative results and performance comparison will be given in chapter 5 where a more accurate model will be introduced in that chapter.

Table 4.2: Overall results of the Prediction model

	Prediction Model	Prediction Model with Rectification
MAE	20.2458	16.2043
standard deviation	8.4238	7.7401

4.6.2 Prediction Model Applied to Partial Input

In this subsection, we design two experiments to verify our algorithm when not all the singular points can be detected. In NIST Database 4, almost all the fingerprint images contain all the singular points correctly. Therefore, it is not suitable for the purpose of these experiments. Instead, we used the FVC2000 Database DA1. The database contains 800 images of 100 fingers, with 8 impressions per finger.

All these 800 images can be grouped into two categories: Category C1 where for each finger, there is at least one image which contains all the singular points present in the finger. Thus we can manually assign a delta point to the images which do not contain all the singular points. Category C2 are for the impressions where for each finger, none of the impressions include all the singular points present in the finger.

Experiments III and IV are conducted according to the existence of the singular points. In experiment III, we want to verify the accuracy and consistency of the proposed algorithm. Thus only category C1 is used as this allows us to determine the performance quantitatively. In experiment IV, we use all the 800 images to verify the performance of the search algorithm using predefined models.

Experiment III: Delta point prediction using search model

(a) Loop model:

For loop model, we design two experiments: one is for the case that all the singular points appear in the image; the other is for the case where the delta point does not appear in the image but can still be determined according to the other images of the

same finger in the database.

In the first case, we select 40 images from the database FVC-2000 DA1. The estimated location of the delta point is compared with that obtained manually. The average standard deviation of the position of delta points are:

$$\sigma_x = 1.78; \quad \sigma_y = 0.73$$

where subscript “x” and “y” refer to the deviation along the x-axis and y-axis respectively.

In the second case, another 40 images from the database FVC-2000 DA1 are tested. The computed average standard deviation of the position of delta points are:

$$\sigma_x = 2.73; \quad \sigma_y = 1.40$$

(b) Whorl model:

Similarly we evaluate the search algorithm using a database of 80 images belonging to the whorl class. All the impressions do not contain any delta points. The estimated average standard deviations of the position of the delta points are:

$$\sigma_x = 2.42; \quad \sigma_y = 2.272$$

The small standard deviations obtained from the various tests conducted show that the search algorithm is able to estimate the position of the delta points with good accuracy and consistency. With this estimation, we can reconstruct the orientation image according to the method presented in section 4.4. Some examples of the reconstructed fingerprint orientation images are shown in Fig. 4.16. It will only

consider 3 types of fingerprint classes because there is no delta point in the plain arch class and the behavior of right loop is similar to that of the left loop and can be considered together.

The use of the search algorithm has its limitation. Error will occur in the following conditions:

1. The impression does not contain sufficient information for effective estimation of its class. Even though the search algorithm does not require the availability of the delta point, it will still require sufficient orientation pattern to estimate its class. Thus significant error will occur if a large part of the fingerprint impression is missing;
2. The central pocket loop has one of the delta points close to the reference point. The proximity between the delta point and the reference point complicates the computation of the orientation image estimation as small location error will cause significant difference in the resultant orientation image;
3. The impression is rotated by more than 30 degrees from normal. Such an error occurs as the assumption made in the formulation of the algorithm does not take into account the orientation of the fingerprint impressions. This can be corrected by doing the search at different orientation angle.

Experiment IV: Search algorithm using predefined models

We evaluate the proposed search algorithm on 800 images in the FVC-2000 DA1 database. According to the number and the position of singular points, the fingerprint images are divided into 4 groups: Arch, Left Loop, Right Loop and Whorl. Thus if the singular points detected by the search algorithm has the similar number and position as the original image, it is assumed that the search algorithm has ap-

Figure 4.16: Examples of the search model: (a) & (b) are for tented arch; (c) & (d) for loop; (e) & (f) for whorl

Table 4.3: Results of search algorithm from 800 images used in the experiment

True Number and Position	detected number and position				Error Rates
	A	L	R	W	
A	142	0	0	2	1.4%
L	1	200	0	7	3.8%
R	25	0	190	25	20.8%
W	8	29	16	164	21.1%

L (Left Loop): 1 delta is right to core; R (Right Loop): 1 delta is left to core

W (Whorl): 2 cores 2 deltas, A (Arch): no delta

proximately predicted the singular points. The result is tabulated in Table 4.3. The average total error rate achieved is 12.7%. Most of the error occurs in the whorl and loop classes. This could be due to the following reasons:

1. For the central pocket loop, when there exists a delta point near to the reference point, its characteristic is quite similar with the loop class. Thus it is difficult to tell them apart. This is illustrated in Fig. 4.17 (a);
2. The upper half impression of the whorl class has orientation characteristic that is quite similar to the loop class. Thus, if a large amount of information is missing in the whorl type, chances of error will be high. This is illustrated in Fig. 4.17 (b).

4.7 Summary

In this chapter, we proposed a new algorithm to model the fingerprint orientation images. The algorithm describe the behaviors of singular points in the fingerprint image using the first order phase portrait model. In order to globally reconstruct the fingerprint orientation, we divided the orientation into several pieces. In each piece, there is only one singular point. We further exploited the other useful characteristic of

(a) A case of central pocket loop

(b) Insufficient input data

Figure 4.17: Error analysis for search algorithm

the orientation image whereby the orientation would not change abruptly except for the singular point region. With this characteristic, the orientation in this region can be described using the extension of the first order phase portrait model of the singular point. Therefore the model can predict the orientation at both the peripheral region and the pattern region. The information used to reconstruct the overall orientation are the positions and behaviors of the singular points. In cases with missing singular point, a search algorithm is developed to predict the position of the missing singular point. Some predefined models with potential singular points are proposed to reduce the search time.

The proposed prediction model has several advantages as follows:

1. The behaviors of singular points can be modeled by the first order phase portrait model. It can precisely reflect the differences among the same type of singular points. In addition we can get the model parameters using only the information from the region near the singular points. At the region away from singular points, the orientations can be predicted by these parameters due to orientation continuity. This may overcome the shortcomings brought by the filter-based local operation.

2. Given all the singular points, the model can reconstruct the overall orientation in piecewise manner. In each piece, the orientation will be decided by its singular point behavior.
3. The proposed search algorithm has the capability to reconstruct the orientation when one or more of the singular points are missing. It is also able to predict the positions of the missing singular points.
4. For incomplete images, we define 37 prediction models in terms of pre-designated singular points. These 37 prediction models can help to improve the effectiveness of the search algorithm.

All the experimental results shown in section 4.6 indicate that our algorithm can work well even if some information is missing in the fingerprint image. The only assumption we make is that we can correctly detect the reference point in the image. Of course, if we lose the reference point in the input image, there will be insufficient for classification or verification.

It is noted that the first order phase portrait model can only describe linear orientation behavior. Thus if the nonlinear variation occurs in the fingerprint image, the predicted orientation may not be correct. In other words, for more accurate description of the detailed orientation images, we should explore more complex models. This will be the main topic in the next chapter.

Chapter 5

Constrained Nonlinear Phase Portrait Model

5.1 Introduction

In the last chapter, we proposed a prediction method to model the fingerprint orientation pattern. However, the orientation obtained using the prediction model are not always correct especially in some local regions away from the singular points. This is because the first order phase portrait model can only successfully describe the linear orientation pattern but failed for some complex patterns.

Several methods have been proposed to improve the estimation of the orientation image. These can be broadly categorized as filter-based [6, 20, 25, 26, 32] and model-based [7, 8, 59–62]. Filter-based methods improve the orientations at a local region but not able to solve large noisy or missing patches in the image. Such methods can be viewed as the local constraint methods. On the other hand, model-based methods consider the global constraint and regularity of orientation image except

for the region near the singular points. Thus, these methods are able to overcome the limitation of filter-based methods. This chapter will focus on the model-based methods.

5.2 Constrained Nonlinear Phase Portrait Model

In the previous chapter, we proposed to represent the behaviors of singular points using the first order phase portrait model. This model gives accurate local description near the singular point but has limited capability to globally describe the orientation images of all the fingerprints. The first order phase portrait model can only deal with the orientation image in which there exists at most one singular point and will fail for loop and whorl where there are more than two singular points.

On the other hand, the nonlinear polynomial phase portrait model has been verified to be able to effectively reconstruct the global orientation of fingerprints in [60]. But in the singular point region, the reconstructed orientation may be wrong.

In order to examine the singular point behavior in the nonlinear phase portrait model, we rewrite Eqn. 2.6 (a & b):

$$dx = m * x + n * y + O_1(x, y) \quad (5.1a)$$

$$dy = c * x + d * y + O_2(x, y) \quad (5.1b)$$

where $O_i(x, y)/r \rightarrow 0$ as $r = (x^2 + y^2)^{1/2} \rightarrow 0, i = 1, 2$, (c, d, m, n) are the first order coefficients.

The linear system

$$dx = m * x + n * y \quad (5.2a)$$

$$dy = c * x + d * y \quad (5.2b)$$

is said to be the linearization (or linearized system) of Eqn. 5.1 (a & b) at the origin. The components of the linear vector field in Eqns. 5.2 (a & b) are said to form the linear part of the high order phase portrait model.

According to the linearization theorem stated in [63]: “Let the non-linear system has a simple singular point at $\vec{x} = 0$. Then in a neighborhood of the origin the phase portraits of the system and its linearization are qualitatively equivalent. Here a singular point at the origin of a non-linear system is said to be simple if its linearized system is simple”, since the singular point in fingerprints has its own complete behavior and can be considered as the simple singular point, the nonlinear phase portrait model can be linearized at the singular point.

By shifting the origin in Eqn. 2.6 (a & b) to the singular point (x_s, y_s) (replace (x, y) as $(x - x_s, y - y_s)$) and expanding it, we obtain Eqn. 5.3 (a & b):

$$\begin{aligned} \cos(2\theta') &= \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} (x - x_s)^{i-j} (y - y_s)^j \\ &= \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} (-x)_s^{i-j} (-y)_s^j + \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} (i-j) (-y)_s^j x + \quad (5.3a) \\ &\quad + \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} j (-x)_s^{i-j} y + O_1(x^p y^q) \quad p + q \geq 2 \end{aligned}$$

$$\begin{aligned}
\sin(2\theta') &= \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} (x - x_s)^{i-j} (y - y_s)^j \\
&= \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} (-x)_s^{i-j} (-y)_s^j + \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} (i-j) (-y)_s^j x + \quad (5.3b) \\
&\quad + \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} j (-x)_s^{i-j} y + O_2(x^p y^q) \quad p+q \geq 2
\end{aligned}$$

where $O_i(x^p, y^q)$ are the high order terms.

Neglecting the high order terms, Eqn. 5.3 (a & b) should have the similar orientation pattern as in Eqn. 5.2 (a & b). Comparing Eqn. 5.3 (a) with Eqn. 5.1 (a) gives the set of constraints for estimating coefficients $\{a_{ij}\}$ using the pre-computed low order coefficients (m, n) for a particular singular point as shown in Eqn. 5.4 (a).

$$\begin{cases} \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} (-x)_s^{i-j} (-y)_s^j &= 0 \\ \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} (i-j) (-y)_s^j &= m \\ \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} j (-x)_s^{i-j} &= n \end{cases} \quad (5.4)$$

and $\{b_{ij}\}$ using (c, d) with constraints in Eqn. 5.5.

$$\begin{cases} \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} (-x)_s^{i-j} (-y)_s^j &= 0 \\ \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} (i-j) (-y)_s^j &= c \\ \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} j (-x)_s^{i-j} &= d \end{cases} \quad (5.5)$$

Incorporating the constraints of the first order phase portrait with local descriptions of the high order phase portrait gives the constrained high order phase portrait model. Thus the proposed model incorporates the best of both the local and global descriptions, which can describe the global orientation image but still preserves an accurate local description near the singular points.

5.3 Solution of Coefficients and Perturbation Analysis

The solution of the coefficients $\{a_{ij}\}$ and $\{b_{ij}\}$ are obtained from the x and y components separately. Thus the perturbation analysis is also done on each component separately. To understand the stability of the solution for either $\{a_{ij}\}$ or $\{b_{ij}\}$, the following analysis takes the $\{a_{ij}\}$ as the example of perturbation analysis. The x component can be represented by its corresponding coefficients $\{a_{(n-m)m}\}$ as shown in Eqn. 5.6:

$$dx(x_i, y_j) = \sum_{n=0}^N \sum_{m=0}^n a_{(n-m)m} x_i^{n-m} y_j^m \quad (5.6)$$

where N represents its order.

Rewrite it in a matrix form as follows:

$$\mathbf{DX} = \mathbf{XY} \cdot \mathbf{A} \quad (5.7)$$

$$\text{where } \mathbf{A} = \begin{bmatrix} a_{00} & a_{10} & a_{01} & \cdots & a_{(n-m)m} & \cdots & a_{0N} \end{bmatrix}^T,$$

$$\mathbf{DX} = \begin{bmatrix} \cos(2\theta(x_1, y_1)) \\ \cos(2\theta(x_1, y_2)) \\ \vdots \\ \cos(2\theta(x_I, y_J)) \end{bmatrix} \quad \text{and} \quad \mathbf{XY} = \begin{bmatrix} 1 & x_1 & y_1 & \cdots & x_1^{n-m} y_1^m & \cdots & y_1^N \\ 1 & x_1 & y_2 & \cdots & x_1^{n-m} y_2^m & \cdots & y_2^N \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x_I & y_J & \cdots & x_I^{n-m} y_J^m & \cdots & y_J^N \end{bmatrix}.$$

The existence of noise in a fingerprint image causes the orientation image to deviate from its accurate value. Instead of not considering the information on the noisy region, a looser weight is associated to this region.

Let $\mathbf{W} = \text{diag}(\omega(x_1, y_1), \omega(x_1, y_2), \dots, \omega(x_I, y_J))$ be a diagonal matrix; $\omega(x_i, y_j)$ denotes the weight in (x_i, y_j) . Multiply $\mathbf{W}$ to both sides of Eqn. 5.7:

$$\mathbf{W} \cdot \mathbf{DX} = \mathbf{W} \cdot \mathbf{XY} \cdot \mathbf{A} \quad (5.8)$$

The weights can be obtained using the coherence value of the orientation as described in Eqn. 5.21.

The constraints in Eqn. 5.4 have its matrix form as shown in Eqn. 5.9:

$$\mathbf{L} \cdot \mathbf{A} = \mathbf{H} \quad (5.9)$$

where $\mathbf{H} = \begin{bmatrix} 0 & m & n \end{bmatrix}^T$ and

$$\mathbf{L} = \begin{bmatrix} 1 & \dots & (-x)_s^{i-j}(-y)_s^j & \dots & (-y)_s^n \\ 1 & \dots & (i-j)(-y)_s^j & \dots & 0 \\ 0 & \dots & j(-x)_s^{i-j} & \dots & n \end{bmatrix} \quad (5.10)$$

Hence the constrained nonlinear phase portrait model can be viewed as the weighted linear least square problem with linear equality constraints, as stated in the following Eqn. 5.11:

$$\min_{\mathbf{A} \in B} \|\mathbf{W}(\mathbf{XY} \cdot \mathbf{A} - \mathbf{DX})\|_2, \quad B = \{\mathbf{A} | \mathbf{L} \cdot \mathbf{A} = \mathbf{H}\} \quad (5.11)$$

where $\|\cdot\|_2$ is the norm.

If the constrained matrix $\mathbf{L}$ has full row rank: $\text{rank}(\mathbf{L}) = p$, the least square problem has the unique solution, which has been proofed by Eldèn [64] (The following

Eqns. 5.12~ 5.17 are based on the perturbation analysis in the work [64]:

$$\mathbf{A} = \mathbf{L}_{I \cdot \mathbf{W} \cdot \mathbf{XY}^+} \cdot \mathbf{H} + (\mathbf{W} \cdot \mathbf{XY} \cdot \mathbf{P}_0)^+ \cdot \mathbf{W} \cdot \mathbf{DX} \quad (5.12)$$

where $(\cdot)^+$ represents the pseudo inverse of a matrix, $\mathbf{L}_{I \cdot \mathbf{W} \cdot \mathbf{XY}^+}^+ = (\mathbf{I} - (\mathbf{W} \cdot \mathbf{XY} \cdot \mathbf{P}_0)^+ \cdot \mathbf{W} \cdot \mathbf{XY}) \cdot \mathbf{L}^+$, and $\mathbf{P}_0 = \mathbf{I} - \mathbf{L}^+ \cdot \mathbf{L}$.

Now let us consider the perturbation problem defined in Eqn. 5.13:

$$\min_{\mathbf{A} \in \tilde{B}} \|\tilde{\mathbf{W}} \cdot \tilde{\mathbf{XY}} \cdot \mathbf{A} - \tilde{\mathbf{W}} \cdot \tilde{\mathbf{DX}}\|, \quad \tilde{B} = \{\mathbf{A} | \tilde{\mathbf{L}} \cdot \mathbf{A} = \tilde{\mathbf{H}}\} \quad (5.13)$$

where

$$\begin{aligned} \tilde{\mathbf{XY}} &= \mathbf{XY} + \Delta \mathbf{XY} \mathbf{E}_{XY} \\ \tilde{\mathbf{DX}} &= \mathbf{DX} + \Delta \mathbf{DX} \mathbf{E}_{DX} \\ \tilde{\mathbf{L}} &= \mathbf{L} + \Delta \mathbf{L} \mathbf{E}_L \\ \tilde{\mathbf{H}} &= \mathbf{H} + \Delta \mathbf{H} \mathbf{E}_H \\ \tilde{\mathbf{W}} &= \mathbf{W} + \Delta \mathbf{W} \mathbf{E}_W \end{aligned} \quad (5.14)$$

and $\|\mathbf{E}_{XY}\| = \|\mathbf{E}_{DX}\| = \|\mathbf{E}_L\| = \|\mathbf{E}_H\| = \|\mathbf{E}_W\| = 1$,

$\Delta \mathbf{XY}, \Delta \mathbf{L}, \Delta \mathbf{DX}, \Delta \mathbf{H}, \Delta \mathbf{W} \geq 0$.

In this study, the matrix $\mathbf{XY}$ and $\mathbf{L}$ are combination of the coordinates of the points (x, y) , so we have $\Delta \mathbf{XY} = 0$ and $\Delta \mathbf{L} = 0$.

According to the perturbation theory for the least square problem with linear equality constraints developed by Eldèn in [64], the estimation errors $\Delta \mathbf{A} = \tilde{\mathbf{A}} - \mathbf{A}$

can be expressed as Eqn. 5.15:

$$\begin{aligned}\|\Delta \mathbf{A}\| &\leq \|(\mathbf{XY} \cdot \mathbf{P}_0)^+\| \|\mathbf{W}^+\| \|\mathbf{DX} - \mathbf{XY} \cdot \mathbf{A}\| \Delta \mathbf{W} \\ &\quad + \|(\mathbf{XY} \cdot \mathbf{P}_0)^+\| \Delta \mathbf{DX} \\ &\quad + \|\mathbf{L}_{I,XY}\| \Delta \mathbf{H}\end{aligned}\tag{5.15}$$

The relative error can be estimated using Eqn. 5.16:

$$\begin{aligned}\frac{\Delta \mathbf{A}}{\mathbf{A}} &\leq \frac{\Delta \mathbf{DX}}{\mathbf{DX}} \cdot \kappa_{\mathbf{L}}(\mathbf{XY}) \frac{\|\mathbf{DX}\|}{\|\mathbf{XY}\| \|\mathbf{A}\|} \\ &\quad + \frac{\Delta \mathbf{H}}{\mathbf{H}} \cdot \kappa^{\mathbf{XY}}(\mathbf{L}) \frac{\|\mathbf{H}\|}{\|\mathbf{L}\| \|\mathbf{A}\|} \\ &\quad + \frac{\Delta \mathbf{W}}{\mathbf{W}} \kappa_{\mathbf{W}} \kappa_{\mathbf{L}}(\mathbf{XY}) \frac{\Delta \mathbf{DX}}{\mathbf{DX}}\end{aligned}\tag{5.16}$$

where

$$\kappa_{\mathbf{L}}(\mathbf{XY}) = \|\mathbf{XY}\| \cdot \|(\mathbf{XY} \cdot \mathbf{P}_0)^+\| \tag{5.17a}$$

$$\kappa^{\mathbf{XY}}(\mathbf{L}) = \|\mathbf{L}\| \cdot \|\mathbf{L}_{\mathbf{I},\mathbf{XY}}^+\| \tag{5.17b}$$

$$\kappa_{\mathbf{W}} = \|\mathbf{W}\| \|\mathbf{W}^+\| \tag{5.17c}$$

We may justify the stability of estimation of the coefficients $\mathbf{A}$ according to the constrained condition numbers $\kappa_{\mathbf{L}}(\mathbf{XY})$, $\kappa^{\mathbf{XY}}(\mathbf{L})$ and $\kappa_{\mathbf{W}}$ defined in Eqn. 5.17 (a & b & c) and its relative error bound according to $\frac{\Delta \mathbf{A}}{\mathbf{A}}$ defined in Eqn. 5.16.

From Eqn. 5.16, the estimation error of coefficients comes from two cases: the original orientation and constraints. If the perturbation of the original orientation is large, which is always indicated by the small coherence value, this will cause large $\kappa_{\mathbf{W}}$, $\Delta \mathbf{DX}$ and $\Delta \mathbf{W}$. Therefore the estimated error of coefficients $\mathbf{A}$ will be large as well. On the other hand, if the constrained matrix $\mathbf{L}$ is ill conditioned, which means

that the $\mathbf{L}$ tends to be singular, it will result in a large $\kappa^{XY}(\mathbf{L})$. Therefore in this case, the estimated error of the coefficients $\mathbf{A}$ may be large also.

In the following two sections, we will discuss the solution for these two cases.

5.3.1 Robustness in Case of a Large Patch of Noise

If the most orientations in the fingerprint image are spoiled by noise, we don't expect to reconstruct such orientation. In this section, we will try to reconstruct the orientation in a fingerprint image where there exists a large patch of noise that affects an enclosed region while out of this region we can still obtain the trustworthy orientation.

In order to solve this problem, we proposed to combine the prediction model with the constrained nonlinear phase portrait model. Generally there are several approaches to compute the quality of a region in an image [6]. In this work, a simple method to segment the good quality image from the noisy region is to use the coherence value. When the coherence value of a block in an image is below a certain threshold, we regard it as an untrustworthy region. The experimental results in chapter 4 show that the orientations in this untrustworthy region are less accurate than the predicted orientations using the prediction model. So we replace the original orientation with the predicted one. However the coherence value assigned to these predicted orientations is small, leading to a small weight value. This is to reduce the impact of the predicted orientation on the original trustworthy orientations.

Figure 5.1: Examples of fingerprint with two close singular points

5.3.2 Robustness in Cases where Singular Points Are Close to One Another

In this section, we will discuss the second case where there are large perturbation of the constraints.

The constraints that will be added to the nonlinear phase portrait shall abide by the linearization theorem. Therefore in fingerprints the two singular points should be well separated so that each singular point exhibits a simple behavior that can be described by the first order phase portrait.

The computation of the orientation uses the pixel values in a block of size $h \times w$ (typically 8×8 or 16×16). Thus the orientation image has a smaller size. In some loop(or tented arch) cases, the core point and the delta point are close to each other in the orientation image. Therefore the singular point region cannot be viewed as a simple singular point. They will interact with each other such that the first order phase portrait description fail near the singular regions. Therefore the perturbation ΔH of $\mathbf{H}$ computed from the first order phase portrait is quite large. Moreover, it will cause a smaller (x_s, y_s) . Let us consider the constraints matrix in Eqn. 5.9. If (x_s, y_s) is small, the elements in the first row will be smaller than the elements in the other rows. This will cause the condition number $\kappa^{XY}(\mathbf{L})$ large, which means the $\mathbf{L}$ tends to be singular. With large $\kappa^{XY}(\mathbf{L})$ and ΔH , the estimated error of the coefficients $\mathbf{A}$ will be large. In Fig. 5.1 (a) we show a fingerprint with two close singular points. Its reconstructed orientation using the first order phase portrait as constraints near singular points is shown in Fig. 5.1 (b). In Fig. 5.1 (c) we also show the reconstructed orientation using the phase portrait model without any constraints. They all fail to precisely reconstruct the orientation at the singular point region.

In order to model the orientation in these cases, we consider the functions 5.18:

$$\begin{aligned} dx &= y \times (y - y_{delta}) \\ dy &= -x \times y_{delta} + y \times x_{delta} \end{aligned} \tag{5.18}$$

and the orientation reconstructed using these functions is shown in Fig. 5.2:

Figure 5.2: Reconstructed orientation using Eqns. 5.18

From Fig. 5.2, we can observe the loop orientation behavior where the core and the delta are close to each other. The solutions to the Eqns. 5.18 are $(x = 0, y = 0)$ and $(x = x_d, y = y_d)$. Supposing the core is at the origin and the delta is at (x_d, y_d) , using Eqn 5.18, we can coarsely reconstruct the loop orientation with constraints of two close singular points.

In order to obtain the more precise reconstructed orientation, we incorporate polynomial functions to Eqn. 5.18 as in Eqn. 5.19 (a & b):

$$dx = (y \times (y - y_d)) \times \left(\sum_{n=0}^N \sum_{m=0}^n a_{(n-m)m} x^{n-m} y^m \right) \quad (5.19a)$$

$$dy = (-x \times y_d + y \times x_d - b \times O^p(y)) \times \left(\sum_{n=0}^N \sum_{m=0}^n b_{(n-m)m} x^{n-m} y^m \right) \quad (5.19b)$$

where $b \times O^p(y)$ is a function to modify the curvature at which the orientation is 0° . This is because the function $dy = (-x \times y_d + y \times x_d)$ will cause the zero-orientation points to lie a line $-x \times y_d + y \times x_d = 0$. At the same time, the item $b \times O^p(y)$ should satisfy the condition $\forall y \rightarrow 0, b \times O^p(y) \rightarrow 0$.

The items $\{\sum_{n=0}^N \sum_{m=0}^n a_{(n-m)m} x^{n-m} y^m\}$ and $\{\sum_{n=0}^N \sum_{m=0}^n b_{(n-m)m} x^{n-m} y^m\}$ are used to correct the orientations other than 0° .

In Fig. 5.1 (d), we show an example of the reconstructed orientation using the constraints in Eqns. 5.19a and 5.19b.

5.4 Implementation of the Proposed Model

In this section, we will describe in detail the proposed algorithm.

1. Compute the original orientation image θ_{orig} and its coherence $Coherence_{orig}$ respectively using the Eqns. 5.20 and 5.21;

$$\theta_{orig}(x_i, y_i) = \frac{1}{2} \arctan\left(\frac{\sum 2G_x G_y}{\sum (G_x^2 - G_y^2)}\right) + \frac{\pi}{2} \quad (5.20)$$

$$Coherence_{orig}(x_i, y_i) = \frac{(\sum (G_x^2 - G_y^2))^2 + 4(\sum G_x G_y)^2}{(\sum (G_x^2 + G_y^2))^2} \quad (5.21)$$

where G_x and G_y are the gradient vectors along the x and y components respectively computed in the window region centered at (x_i, y_i) .

2. Compute the predicted orientation using the proposed prediction model described in Chapter 4:
 - (a) For each located singular point, the first order phase portrait described in chapter 4 will be used to reconstruct its orientation.
 - (b) In terms of the number and the position of singular points, the rules in Chapter 4 are adopted to form the predicted orientations θ_{pred} for the whole image.

3. Form the training data θ_{trn} by combining the original orientation θ_{orig} and the predicted orientation θ_{pred} according to the following criteria:

$$\theta_{trn} = \begin{cases} \theta_{pred} & \text{if } Coherence_{orig} < 0.5 \\ \theta_{orig} & \text{if } Coherence_{orig} \geq 0.5 \end{cases}$$

and

(5.22)

$$Coherence_{trn} = \begin{cases} 0.3 & \text{if } Coherence_{orig} < 0.5 \\ Coherence_{orig} & \text{if } Coherence_{orig} \geq 0.5 \end{cases}$$

Note that once the original orientation is replaced with the predicted orientation, the coherence should also be changed to a value $0 < Coherence_{trn} < 0.5$.

We use a value of 0.3 which we found suitable through trial and error.

4. Compute the orientation model using the constrained nonlinear phase portrait algorithm.
- (a) Compute the first-order phase portrait at each singular point region as the constraints using Eqn. 5.4 and 5.5;
 - (b) Prepare the training data obtained in step 3 along the x and y components respectively;

$$\begin{aligned} dx &= \cos(2 * \theta_{trn}(x, y)) \\ dy &= \sin(2 * \theta_{trn}(x, y)) \end{aligned}$$
(5.23)

- (c) Compute all the coefficients of the nonlinear phase portrait a_{ij} and b_{ij} .

For each singular point, three constraints will be given by Eqn. 5.4 for a_{ij} and Eqn. 5.5 for b_{ij} . Thus a total of $3n$ (where n is the number of the singular point) coefficients will be reduced from Eqn. 2.6. Weighted least square algo-

rithm is used to compute the coefficients a_{ij} and b_{ij} so that the least square errors between training data θ_{trn} and the phase portrait model is minimized. Coherence matrix will act as the weighting factor when determining the overall orientation value from the actual, predicted and the computed value of the model.

Fig. 5.3(a–f) show the resultant images obtained in each step.

Figure 5.3: Resultant images obtained in each step

5.5 Experiments

Experiment I: Quantitative analysis

In Experiment I, we evaluate our proposed algorithm on the same images as those used in Experiments II of Sec 4.6 (500 NIST DB4 images). We set the order of nonlinear phase portrait to 8 as a trade-off between accuracy and conciseness of the model. The MAE of the model is 9.4025° with a standard deviation (std) of 5.8084° .

Table 5.1: Comparison results of the different orientation models

	Zero Pole Model	Piecewise non- linear Model	Combina- tion Model	Prediction Model	Constrained non linear Model
MAE	21.5556	15.2376	11.7357	16.2043	9.4025
std	8.3821	6.8435	4.2698	7.7401	5.8084

std: standard deviation;

As the comparisons, we list the MAE with the corresponding std using the proposed constrained nonlinear phase portrait model, our earlier prediction model and the reported approaches of Zero-pole model, linear piecewise zero-pole model and combination model. The most accurate method is the constrained nonlinear phase portrait model. This is followed by combination model, linear piecewise zero-pole model, prediction model and the zero-pole model in the descending order. This order is reasonable. The zero-pole model only considers the positions of the singular points. Thus it can only give constant description of the orientation as long as the positions of the singular points are known. The prediction model and the linear piecewise zero-pole model improve the estimation of the orientation by certain linear rectification. However, they cannot represent the nonlinear changes of the orientation images. Compared with the combination model, the constrained nonlinear phase portrait model integrates the description of the singular points into the high order

phase portrait model. Therefore, the constrained nonlinear phase portrait model is able to describe both the local orientations near the singular points and the global orientations. By incorporating the prediction model to remove the unreliable orientation in the noisy regions, the constrained nonlinear phase portrait model is more robust to noise than the combination model.

Experiment II: Qualitative analysis

For fingerprint images that do not contain full ridge information, it is difficult to evaluate the effectiveness of the algorithm quantitatively. Here we give some resultant images with its reconstructed orientation using the proposed model to show the effectiveness of our algorithm qualitatively. In Fig 5.4, we give four examples, each belonging to a different fingerprint type: plain arch, tented arch, loop and whorl respectively. In these examples, the orientation images are given in the region where there is clear ridge information, as well as in the region without clear ridge information, such as in the peripheral areas.

Experiment III: Experimental results for cases of fingerprints containing a large noise patch

Since the predicted orientation images are computed using only partial data (around the singular points), our proposed method can be used to replace the orientation with weak coherence value, especially at areas with large noisy patch. In Fig. 5.5, a comparative result is given between the proposed model and the combination model which is chosen as it is the most accurate approaches in the prior art as shown in Table 5.1. Fig. 5.5 (a) shows an original fingerprint image with a large noisy patch at the left-bottom section of the image. Fig. 5.5 (b) shows its original orientation image computed. Fig. 5.5 (d) shows the reconstructed orientation using the combination model approach. The combination model cannot provide a correct

Figure 5.4: Examples of reconstructed orientation

solution for the large noisy area as it only uses the weights as the constraints when estimating the coefficients. Thus when the noisy area is large, the error will be large as well. Our proposed model considers the prior knowledge and uses the global constraints to predict the orientation with weak coherence value. This can correct the orientation in a large noisy area. The result shown in Fig. 5.5 (c) is the reconstructed orientation using our proposed model, which is qualitatively better than the result

(a) The original image

(b) The original orientation image

(c) Reconstructed orientation using the proposed model

(d) Reconstructed orientation using the combination model

Figure 5.5: Comparative results between the proposed model and the combination model

in Fig. 5.5 (d).

Two more examples for reconstructed orientation from poor quality fingerprint images with a large patch of noise are illustrated in Fig. 5.6 (a–d). Though the middle part of Fig. 5.6 (a) and the left upper part of Fig. 5.6 are damaged by a large noise patch, our proposed method can still correctly reconstruct their orientation.

(a) A loop image with a large noisy patch

(b) Reconstructed orientation of (a) using the proposed method

(c) A whorl image with a large noisy patch

(d) Reconstructed orientation of (c) using the proposed method

Figure 5.6: Examples of the reconstructed orientation of poor quality image using the proposed model

We test our algorithm on the first 500 images of the NIST database 4 ($f0001 - f0250$ and $s0001 - s0250$). For most of the images, our method can correctly model the orientation images. However for 30 images, the reconstructed model cannot be formed properly. These images do not have sufficient good orientation information around the singular point regions and thus the algorithm cannot extract the correct singular point behavior. In such cases, the prediction model cannot be obtained

properly. An example of an error image is shown in Fig. 5.7. If we fail to extract the singular point in the image due to poor quality, the proposed model will fail; similarly for the combination model.

(a) A poor quality image

(b) Reconstructed orientation using the proposed model

Figure 5.7: Example of a poor quality image where the reconstruction fails

5.6 Summary

In this chapter, we proposed a constrained nonlinear phase portrait model to reconstruct the fingerprint orientation. The first order phase portrait model, which is suitable to describe the local region, cannot precisely represent the complex orientation. On the contrary, the nonlinear phase portrait model which has superiority to depict the complex global orientation pattern, may fail at the singular point region. Therefore, we use the first order phase portrait model to reconstruct the orientation images near the singular points. Subsequently this model is used to constrain the coefficients of the high order phase portrait which is used to model the orientation

pattern of the entire fingerprint. Thus we incorporate the first order phase portrait model into the high order phase portrait model, taking the best of both linear and nonlinear phase portrait models. Finally we obtain a concise and unified orientation model. No ad-hoc weighting is needed to combine the results from multiple models to form the entire fingerprint orientation model.

In order to improve the robustness of the constrained nonlinear model, we make use of the prediction model mentioned in chapter 4 to replace the unreliable orientation in a noisy image before we obtain the constrained nonlinear model. The prediction model is based on the analysis of the regularity of the fingerprint orientation which provides at least a right trend of the orientation image. The experimental results in chapter 4 have already shown its effectiveness. Accordingly, the prediction model improves the robustness of the constrained nonlinear phase portrait model, especially in a noisy image. The experimental results obtained validate the effectiveness of the proposed algorithm.

In summary, the constrained nonlinear phase portrait model has the following advantages:

1. It can reconstruct the global orientation pattern and the local orientation pattern near the singular points simultaneously;
2. It has the concise and unified form with $N * (N + 1) - 2$ coefficients, here N denotes the order of the nonlinear phase portrait model;
3. It is robust even in a noisy image if the prediction model is used as preprocessing, so long as a reference point is correctly detected;
4. It doesn't require all the singular points to appear in the image;

It should be noted that although the constrained nonlinear model does not require

all singular points to be present in the image, it still requires that all the singular points present in the image to be correctly detected. This problem is also faced by all other model-based methods. If all the singular points that appear in the image are not detected correctly, the reconstructed orientation near the singular point may be incorrect.

Chapter 6

Fingerprint Classification Based On Nonlinear Phase Portrait Model

6.1 Introduction

Fingerprint biometrics are increasingly being used in the commercial, civilian and financial domains. The demand varies across a wide range of applications from law enforcement, banking, physical access control, time-and-attendance to consumer products. These result in a large volume of fingerprints being collected and stored in the database. For example, there are more than 81 million fingerprints currently in the FBI Fingerprint database, and every day approximately 7,000 new individual records are added to the files¹. To perform fingerprint recognition, both recognition accuracy and processing time are critical performance issues. In order to achieve an efficient identification, fingerprints in the database are usually organized into a

¹<http://www.fbi.gov/hq/cjisd/ident.pdf>

number of sub-groups sharing similar properties. Therefore we only need to compare a given query fingerprint with those in one single sub-group. This will reduce the processing time significantly. In general, there are two ways to assign fingerprints into the different sub-groups: one is to make use of the prior knowledge related to the individual such as name, gender, race etc. . This approach is effective and accurate but not always available (e.g., looking for a suspect in a criminal scene). The alternative approach is to make use of the fingerprints' intrinsic characteristics to assign a fingerprint into a predefined class. This is known as fingerprint classification.

The investigation of fingerprint classification can be traced back to 1823 when Purkinje classified fingerprints into nine categories: transverse curve, central longitudinal stria, oblique stripe, oblique loop, almond whorl, spiral whorl, ellipse, circle and double whorl [65]. Subsequently Francis Galton proposed a fingerprint classification scheme, where fingerprints were classified into three major classes and further divided each class into subcategories in 1892 [66]. In 1900 Edward Henry refined Galton's method by increasing the number of classes [67]. Today, most of classification schemes were derived from the Galton-Henry classification scheme. In Table 6.1, an example of the Galton-Henry classification scheme is listed. This scheme is adopted by FBI [58].

Table 6.1: FBI fingerprint classification scheme

ARCH	LOOP	WHORL
a. Plain Arch	a. Left Loop	a. Plain Whorl
b. Tented Arch	b. Right Loop	b. Central Pocket Loop
		c. Double Loop
		d. Accidental Whorl

These 8 classes are not uniformly distributed in nature. In particular, the central pocket loop, double loop and the accidental whorl are rare. Therefore for most automated classification systems [1,2,4,20,27,51,68–72], researchers generally adopt

Figure 6.1: Examples of five-class scheme

the more popular 5 classes, namely, plain arch (A), tented arch (T), left loop (L), right loop (R) and whorl (W) as shown in Fig. 6.1. In this work, 5 classes are adopted. Their definitions are provided in [5] as:

- An arch fingerprint has ridges that enter from one side, rise to a small bump, and go out the opposite side from which they entered. Arches do not have loops or deltas;
- A tented arch fingerprint is similar to the plain arch, except that at least one ridge exhibits a high curvature and one loop and one delta are present;
- A loop fingerprint has one or more ridges that enter from one side, curve back, and go out the same side they entered. A loop and a delta singularities are present; the delta is assumed to be at the south of the loop. Loops can be

further subdivided: loops that have ridges that enter and leave from left side are called left loops and loops that have ridges that enter and leave from the right side are called right loops;

- A whorl fingerprint contains at least one ridge that makes a complete 360 degree path around the center of the fingerprint.

Fingerprint classification is a difficult task due to the following two reasons:

- The small interclass variability and large intraclass variability (Fig. 6.2);
- The presence of noise (Fig. 6.3).

Figure 6.2: The first row shows three fingerprints with different types but similar behavior (small interclass variability); the second row shows three fingerprints with the same type with different behaviors (large intraclass variability).

Orientation image plays an important role in fingerprint classification. However, the traditional method of computing the orientation image only considers the orientation information in a small neighborhood region and does not take the global

Figure 6.3: Examples of noisy images

pattern of the fingerprint into consideration [1, 26, 32, 50, 51]. Moreover, the local orientation image may be unstable due to noisy or poor quality images.

Recognizing that the orientations of a fingerprint will only change abruptly near the singular point regions, we developed a constrained nonlinear phase portrait model to globally reconstruct the orientation image in chapter 5. Our work in this chapter are motivated by the following: Since using the concise representation of an orientation model can produce a good reconstruction of the orientation image, it should also be able to perform fingerprint classification. To evaluate the effectiveness of performing fingerprint classification based on the constrained nonlinear orientation model instead of the original orientation image, we proposed an algorithm which uses the coefficients of the constrained nonlinear orientation model combined with the singularities information as features. The details of the algorithm and experiments are discussed in the sections following the literature review of fingerprint classification.

6.2 Literature Review on Fingerprint Classification

In the last 30 years, although fingerprint classification has been extensively investigated and many different methods had been proposed, limited features and classifiers were used to perform the classification. An overview of fingerprint classification according to the features are given:

Singularities

There are two kinds of singular points in the fingerprint image, core point and delta point. A core point is the turning point of an innermost ridge and a delta point is a place where two ridges running side-by-side diverge. According to the position and number of singular points, a fingerprint image can be classified into one of the classes as stated in Table 6.2.

Table 6.2: Classification based on singularities

Classes	cores	deltas	Core's position to delta's
Whorl	2	2	-
Right Loop	1	1	Left
Left Loop	1	1	Right
Tented Arch	1	1	Beneath
Plain Arch	0	0	-

The algorithms based on singularities include:

- Kawagoe and Tojo [49] in 1984. They used the number of singular points to classify the fingerprint into arch, loop and whorl types.
- Karu and Jain [51] in 1996. In their method, $|\frac{1}{n}\sum_{i=1}^n \sin(\alpha_i - \beta)|$ was defined to differentiate the tented arch from loops if the value was below a certain threshold, where $\{\alpha_i\}$ is the local direction angles on the line connecting the

core and the delta points and β is the slope of this line.

- Other singularities based methods could be found in [2, 4, 73–77].

The classifiers in most of the singularity-based methods were rule-based while in Mohammed *et al.* [77] and Nyongesa *et al.* [2], the classifier is based on neural networks.

Singularities provide a simple method to perform classification. However, not all the singular points are available in the fingerprint image.

Ridge Information

Ridge information is another common feature used for fingerprint classification. The following lists the methods that utilize ridge information.

- Kawagoe and Tojo [49] in 1984. After a coarse classification by singularities, the different features were extracted using the ridge flow tracing to further classify whorls into double loop and whorl, loops into left loop, right loop and pocket and tented arch.
- Candela *et al.* [68, 78] in 1994 and 1995. A pseudo-ridge tracing was used to improve the accurate classification of the whorl type.
- Chong *et al.* [79] in 1997. They represented the ridges with B-spline function. Then the similarities between two features were computed to perform the classification.
- Senior [80] in 2001. In this method, a fingerprint was first thinned into one pixel width. Thereafter, two feature vectors were formed. The first feature was generated by the intersection points of fiducial lines and the thinned ridges. The

other feature consisted of key points of a thinned ridge. HMM and decision trees were adopted as the classifiers for these two features.

- Jain and Minut [81] in 2002. The pseudo-ridge tracing was used to extract features which was being compared with the predefined kernels to decide its attribute class.
- Chang and Fan [82] in 2002. A series of ridge patterns were defined in this approach. Then a query fingerprint was described according to the existence and position of these ridge patterns. The sequences of these ridge patterns then determined which class the fingerprint belonged to.
- Shah and Sastry [70] in 2003. In this method, they first obtained a thinned ridge with which multiple oriented points would be preserved. Then the thinned ridge plane was divided into several sections. In each section, the two major orientations were extracted as features. The features were then verified with support vector machine, neural networks and the k-nearest neighbor classifiers.
- Zhang and Yan [4] in 2004. They performed fingerprint classification in terms of singularities first, thereafter a pseudo-ridge tracing was used to improve the classification results. This was a rule-based classification.
- Dass and Jain [83] in 2004. In this approach, the salient features were exploited after ridge tracing. The salient features were the number and locations of the sign-change point as well as the number and locations of local maximums and minimums.
- Xu *et al.* [84] in 2005. Here ridge tracing was applied to extract the points with the maximum curvature as feature points.

The classifiers used for ridge information include: syntactic approach [82], rule-based approaches [4, 49, 79, 81, 83, 84], neural network based approaches [68, 78, 83], structural approaches [80, 85] and statistical approaches [70].

Gabor Filter Responses

In 1999, Jain *et al.* [1] proposed a novel method based on Gabor filters. The Gabor filters were described as follows:

$$G(x, y; f, \theta) = \exp \left\{ \frac{-1}{2} \left[\frac{x_1^2}{\delta_x^2} + \frac{y_1^2}{\delta_y^2} \right] \right\} \cos(2\pi f x_1)$$

where $x_1 = x \sin \theta + y \cos \theta$, $y_1 = x \cos \theta - y \sin \theta$. f is the frequency of the sinusoidal plane wave along the direction θ from the x-axis, and δ_x and δ_y specify the Gaussian envelope along the x and y axes respectively.

For a given query image, Gabor filters were first applied around the core point. The fingerprints were then separated into 48 sections. In each section with the 4 different θ values, the standard deviation was computed as feature vectors, called “FingerCode”. Then it was fed into a k-nearest neighbor classifier and a neural network classifier.

Orientation Image

Orientation image can be interpreted as the direction of the ridge flow. The following algorithms are based on the orientation image of the fingerprints.

- Moayer and Fu in 1975 [86] and 1976 [87]. The features were extracted from the orientation images. Through the analysis of the orientation, the orientation image can be encoded into 12-bit binary number, called “containment code C”. Thereafter, a syntactic approach was used to perform the classification.
- Candela *et al.* [68, 78]. In their method, 1690 orientations in the orientation

image were reduced to 64 principle components by KL transformation. A neural network was then employed as classifier using these 64 features.

The similar algorithms could be found in Halici and Ongun [69], Kamei and Mizoguchi [88], Kamijo [89], Bernard *et al.* [22].

- Cappelli *et al.* [3,90]. The authors represented a fingerprint pattern as a combination of several subspaces. For each subspace, KL transform was applied to extract the features from the orientation image. Multiple KL features were then formed for fingerprint classification.
- Cappelli *et al.* [72,90,91] and Maio & Maltoni [21] and Yao *et al.* [72]. In these papers, the authors reported a method to partition the orientation image by grouping the “homogeneous” orientation image regions using MASKS and dynamic MASKS . This could be viewed as a compact representation of the orientation image. The relational graphics were then formed as the features.
- Neuhaus and Bunke in 2005 [85] used directional variance as the features. The directional variance was defined as $\sigma_{x,y}^2 = \frac{1}{1-n} \sum_{i,j} \sin^2(\alpha_{i,j} - \bar{\alpha}_{x,y})$, where $\alpha_{i,j}$ denoted the vector at position (i, j) of the orientation image. $\bar{\alpha}_{x,y}$ gave the average orientation in the local window around position (x, y) . The classification was then performed with a graph edit distance algorithm.

Other features

There also exist methods of fingerprint classification employing other features in the literature.

- Fitz & Green [92] in 1996 and Park and Park [93] in 2005 used Fourier Transformation as features to perform the classification.

- In 2000, Su *et al.* [94] presented an algorithm based on fractal analysis. The image is first divided into 4 sub-images. In each sub-image, 4 directional fractal parameters, compared with the predefined mask, were calculated.
- Pattichis *et al.* proposed a method which used FM component to enhance the fingerprint prior to classification in 2001.

In Table 6.12, we show the summary of fingerprint classification algorithms available in the literature.

6.3 Feature Extraction

As the most common used feature, orientation image has its own advantages for fingerprint classification. However, the difficulties of the orientation image based methods lie in the following three reasons:

1. There are small interclass variabilities of orientation pattern (which make their orientation pattern similar), examples are shown in the first row of Fig. 6.2;
2. There are large intraclass variabilities of orientation patterns (which make the similarity measurement big), examples are shown in the second row of Fig. 6.2;
3. Noise may cause a large patch of wrong orientation.

In the prior sections of this thesis, we have shown that the nonlinear phase portrait model can produce a better reconstructed orientation than the gradient-based methods. As a result, we postulate that the coefficients of the nonlinear phase portrait model can extract robust orientation information and represent them in a concise format. Thus it has a capability to overcome the third reason which may affect the classification accuracy.

As for the first two circumstances, the singularities-based methods provide more convenient way to classify fingerprints. The difficulties of singularities-based methods lie in the following reasons:

1. The singular points may not appear in the query image;
2. The noise in the fingerprint images makes the singular point extraction unpredictable (missing or wrong detection).

Compared with the singularity-based methods, the orientation image does not require the information of the singular points. Therefore it can tolerate partial images where the singular points may not be included.

Since the advantages of both features compliment each other, it can be expected that using orientation model together with the singularities information will produce an improved classification accuracy compared to the approach that relies on a single feature.

6.3.1 Relationship between Detection of the Singular Points and Computation of Orientation Image

The general methods to detect singular points are the Poincare-based method [49], intersection-based method [52] or filter-based method [54] and phase portrait method in chapter 3. In all these methods, singular points are extracted from the orientation image. Therefore in singular point extraction, the orientation image is considered trustworthy. On the other hand, if given all the singular points information, the orientation image is able to be reconstructed using the model-based methods [7, 8, 59, 95]. In such computation, singular points are considered trustworthy. In both cases, error in the information causes error in the results. To reduce the noise from

the orientation image, a lowpass filter is deployed. Such filtering operation can reduce the noise but risk smoothing the orientation of the singular point regions. Hence the orientation pattern of singular points may disappear. Some noise will cause false detection of singular points. If singular points are wrongly detected, the wrong orientation patterns will be brought into the reconstructed orientation patterns. In most existing computation of singular points and orientation images, there is no consideration of such interaction.

In chapter 3, we have proposed to detect the singular points using phase portrait approach which has considered such interaction. While in this chapter, we will adopt the filter-based approach [54] to detect the candidate singular points. This is because filter-based approach can extract the position and direction of the singular point simultaneously, which is able to align the fingerprint image in the following stage. Moreover filter-based approach can be implemented faster than phase portrait based method due to the separability of the symmetrical filters.

Unlike the works in [54], more than two cores or deltas are extracted as the magnitudes of singular points may not be large enough for a pre-determined threshold as proposed in [54], such as in noisy regions. Nevertheless, their magnitudes may still be relatively large compared to the magnitudes from the other regions. These candidate singular points are then used to reconstruct the orientation image using the methods in [95]. The verification of the candidate singular points is achieved by computing the sum of difference between the reconstructed orientation and the original orientation with high confidence level near these candidate singular points. The use of both orientation and singularities to validate one another can be viewed as an interactive validation mechanism. The best result from both the singular points and the orientation image are selected to form the final feature vectors for classification. In order to reduce the dimensionality of the features used, the coefficients of the

orientation model are used to represent the orientation image. We postulate that the coefficients of the orientation model contains information as rich as the orientation image as it can be used to correctly reconstruct the orientation image.

To compliment the orientation information, the information of singularities used in this work comprises the number of singular points and their relative positions.

6.3.2 Complex Symmetrical Filters Based Method to Detect the Candidate Singular Points

Complex symmetrical filters were proposed by Bigun, Bigun and Nilsson [54, 96, 97] to detect the singular points. The method has been introduced in section 3.2. In works [54], the authors detect the existence of singular points at a coarse level, then a multiscale approach is used to precisely locate singular points. They also assumed that singular points are near the center of image. As such, Gaussian filtering, performed on the entire orientation image, is used to suppress the filters' response at the boundary region of the orientation image.

The complex symmetrical filters are successful used to detect singular points around the center of an orientation image. Its advantages are: 1. The filters can be separated into two 1-dimensional filters, so the computation is effective; 2. The filters can extract singular points (according to the magnitude of filters' response) and their corresponding directions (according to the angle of the filter' response) at the same time. Consequently, the alignment of the image can be done using the position and angular information of the detected singular points.

Their works assume that the landmark points are always in the center region of the fingerprint image. Therefore it is hard to detect the singular points near the boundary of a fingerprint image. So the proposed method is modified to suit the need

of this work which aims to detect all the existing singular points. For example, the threshold in their work is set high, which can ensure that the detected singular point is a real singular point. However, a high threshold could result in the mis-detection of singular points in noisy regions or due to the suppression done at the boundary of the orientation image. In fact, the filters' response provides a continuous measurement of the orientation patterns. At the local regions around the singular points, the filters responses always take the local maximum values. Therefore, instead of setting a fix threshold, the points with the local maximum magnitude of the filters' response are extracted as the candidate singular points.

In this work, the use of complex symmetrical filters is to extract, as much as possible, all the candidate singular points which yields the local maximum magnitudes even if these singular points lie at the boundary of the orientation or is partially corrupted by noise. The angles of the corresponding points are also extracted as the direction of the candidate singular points. The orientation information is useful as the point with the maximum magnitude of core-type filter is used as the landmark point to align the orientation image. The normalized image makes the features from the orientation image translation and rotation invariant.

Using the local maximum criteria, more than 2 cores or deltas can be extracted. However, we are faced with a new dilemma of choosing the true singular points from the list of candidate singular points. One way to verify the singular points detected is to reconstruct the orientation using these detected singular points and to compare it with the original orientation image at the singular points region. In the following part, we will restate the constrained nonlinear phase portrait model of fingerprint orientation.

6.3.3 Constrained Nonlinear Phase Portrait Model of Fingerprint Orientation

The basic idea of the model-based fingerprint orientation computation is to reconstruct each orientation in the entire orientation image using all the original orientation information. The details of the constrained nonlinear phase portrait model has been proposed in chapter 5. Here a brief overview is given.

The orientation image comprises the local orientation behavior near singular points and the global orientation behavior. The mathematical description of the global behavior using the high order phase portrait is given as Eqn. 6.1 (a,b) with the origin shifted to singular points (x_s, y_s) :

$$\begin{aligned} \cos(2\theta') = & \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} (-x_s)^{i-j} (-y_s)^j + \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} (i-j) (-y_s)^j x_s + \\ & + \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} j (-x_s)^{i-j} y_s + O(x^p y^q) \quad p+q \geq 2 \end{aligned} \quad (6.1a)$$

$$\begin{aligned} \sin(2\theta') = & \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} (-x_s)^{i-j} (-y_s)^j + \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} (i-j) (-y_s)^j x_s + \\ & + \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} j (-x_s)^{i-j} y_s + O(x^p y^q) \quad p+q \geq 2 \end{aligned} \quad (6.1b)$$

where $O(x^p, y^q)$ are the high order terms.

The coefficients a_{ij} and b_{ij} of Equation 6.1 should satisfy the following equa-

tions 6.2.

$$\left\{ \begin{array}{l} \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} (-x)_s^{i-j} (-y)_s^j = 0 \\ \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} (i-j) (-y)_s^j = m \\ \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} j (-x)_s^{i-j} = n \\ \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} (-x)_s^{i-j} (-y)_s^j = 0 \\ \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} (i-j) (-y)_s^j = c \\ \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} j (-x)_s^{i-j} = d \end{array} \right. \quad (6.2)$$

where (m, n, c, d) are the coefficients of the first order phase portrait, which are used to control the behavior of the singular points.

The coefficients of the constrained nonlinear phase portrait model can be used to reconstruct the global and local orientation at the same time if the singular points are detected correctly. Therefore, they contain enough orientation information as the original orientation for fingerprint classification. However, in practice, correct detection of the singular points is a difficult task. Since the constrained nonlinear phase portrait model is used to reconstruct the whole orientation image using the singular points information, if the any of the singular point is wrongly detected, the reconstructed orientation would be wrong as well, especially at the wrongly detected singular point region. This gives us an idea to verify singular points using the orientation model. In the next subsection, an interactive validation of singular points and orientation model is proposed.

6.3.4 Interactive Validations of Features

Based on the approach proposed in Sec. 6.3.2, as much as possible, all the possible candidates of the singular points are extracted. Then, from Sec. 6.3.3, the recon-

structed orientation can be obtained using the constrained high order phase portrait after correct detection of the singular points.

Thus the singular point information is useful for obtaining a correct unified orientation model. Vice versa, the reconstructed orientation can be used to check the validity of the detected candidate singular points. Such approach can be viewed as an interactive validation.

The basic idea of the interactive validation approach is that we use the singular points information detected by the complex filters to obtain the phase portrait model and reconstruct the orientations. On the other side, we use the phase portrait model and the reconstructed orientation to help select the true singular points.

For a specific candidate singular point, we define 4 features to select the true singular point.

1. The magnitude of the complex symmetrical filters' response; the different types of the local orientation structures have the different filters' response. The true singular point has generally a big response to the corresponding type of filter.
2. The error (or difference) D between the reconstructed orientation and the original orientation (defined in Eqn. 3.27); This feature is based on the fact that for a real singular point, the difference should be smaller than 0.15.
3. The mean orientation coherence; This value is used to indicate the quality of images and the singular point will be rejected if the value is low.
4. The consistency of the type of the candidate singular points. Recalled from chapter 3, the phase portrait model can provide the type of the singular point. This type should be as the same as the type decided by the complex filters. For the real singular point, these two types should be the same. Otherwise, the

Figure 6.4: An example for the interactive validation of the singular points. (a) all the possible candidates of the singular points obtained using the complex filter approach; (b-f) the reconstructed orientations using different combinations of the candidate singular points; (g) the orientation error, D , of all the candidates singular points; (h) the final reconstructed orientations using four singular points. The core candidates are marked as the white rectangles; the delta candidatures are marked as the black rectangles.

singular point is regarded as spurious.

In Fig. 6.4, an example is given to show the process of interactive validation of the singular points. In this example, a complex filter based method is used to

extract all the possible candidate singular points. A threshold, T_s is set to limit the number of candidates to a reasonable range (in this thesis, a value of 0.45 is used). There are 4 core candidates and 2 delta candidates having filters' response magnitudes greater than the preset threshold T_s . Fig. 6.4 (a) shows these candidate singular points with their corresponding response marked on the image, (b-f) show the reconstructed orientations using different combination of the candidate singular points; (g) shows the orientation error, D , between the reconstructed orientations and the original orientations for all the singular points. Two of the core candidates have the value D greater than 0.15 and are thus regarded as spurious singular points and subsequently rejected. The final reconstructed orientations and the core points (marked as white rectangles) and the delta points (marked as black rectangles) are shown in Fig. 6.4 (h).

6.3.5 Form the Feature Vectors for Fingerprint Classification

After verifying the singular points, the feature vectors can be extracted. As stated earlier, the feature vectors for fingerprint classification in this thesis comprises the orientation information and the singularities information. The orientation information is obtained from the coefficients of the orientation model. In order to make the feature vectors translation and rotation invariant, preprocessing is necessary. The next section describes the implementation detail of the proposed approach.

6.4 Implementation of the Proposed Method

Based on the orientation and singularity information, fingerprint classification can be performed. The overall approach can be divided into 7 stages. This section will describe the overall approach in detail.

Stage 1: Compute the original orientation image and the coherence matrix

The original orientation image θ_{orig} and its coherence $Coherence_{orig}$ are estimated respectively using the Eqns. 5.20 and 5.21;

Stage 2: Extract the candidate singular points

There are two kinds of singular points: core point and delta point. The extraction of the candidate core points is illustrated in this stage. The extraction of the candidate delta points is similar except that the filters adopted is h_d instead of h_c .

Let $z = \cos(2\theta_{orig}) + i \sin(2\theta_{orig})$ represents the complex double orientation image, thus the filter response Rf is shown in Eqn. 6.3:

$$Rf = z * h_c \quad (6.3)$$

where “*” denotes the convolution operator.

Since the candidate core points will always have strong filter response, all points with the local maximum magnitude of filter responses are extracted. To keep the number of candidate singular points to a reasonable number, the local maximum magnitude of filter response should be larger than a pre-determined threshold. Otherwise, the points will be removed from the candidate singular points. Note that the threshold value is not restrictive as it merely determines the processing time needed

for stage 3.

Stage 3: Verification of the candidate singular points

$$\cos(2\theta_{orig}) = 0 + a_{10}x + a_{01}y + \cdots + a_{(i-j)j}x^{i-j}y^j + \cdots + a_{0n}y^n; \quad (6.4a)$$

$$\sin(2\theta_{orig}) = 0 + b_{10}x + b_{01}y + \cdots + b_{(i-j)j}x^{i-j}y^j + \cdots + b_{0n}y^n; \quad (6.4b)$$

For each candidate singular point, the constrained nonlinear phase portrait model is represented using Eqn. 6.4:

Here the constant terms for both x and y components are set to 0, which force the candidate point as a singular point. Then the coefficients $\{a_{i-j}j\}$ and $\{b_{(i-j)j}\}$ can be estimated using the least square algorithm.

From these coefficients, the following information can be obtained: 1. Reconstructed orientation near this point; 2. The type of this point by analyzing the first order coefficient matrix $\begin{bmatrix} a_{10} & a_{01} \\ b_{10} & b_{01} \end{bmatrix}$

Combining the coherence and magnitude of filter response information with the difference between the original orientation and the reconstructed orientation and the type information, the real singular point can be verified.

Stage 4: Alignment of the orientation image

To achieve translation and rotation invariance, it is necessary to align and crop all the fingerprint images to a canonical size and orientation. From the response of the core-type filter, the position and orientation of the primary core (corresponding to the maximum certainty of the core-type filter response) can be obtained. Taking the primary core as the origin, the image is rotated by $(90 - \text{core_orientation})$ degrees to align the core orientation into vertical orientation. Fig. 6.5(a) & 6.5(b) show an

(a) Original fingerprint image with the detected singular points marked

(b) Fingerprint image after alignment

(c) Fingerprint image after cropping

(d) Reconstructed orientation overlaid on the cropped fingerprint image

Figure 6.5: Results for each step in the preprocessing stage

example of a fingerprint image before and after alignment. The alignment ensures that all the images used for the orientation modeling have consistent translation and orientation.

The fingerprint images acquired from the sensor will usually contain noise in the background region. If the entire image is used, the orientation model generated will

be affected by the noise and background region. However, care must be taken such that the cropped image will contain critical information necessary for the classification and at the same time, removing as much background region as possible. For example, a whorl type fingerprint might possess singular points that are separated far apart. In this case, the cropping process should preserve all the singular points as much as possible. After an extensive study, we found that a cropped image of size 320 by 320 pixels is reasonable for fingerprint images obtained using sensor with 500dpi resolution. Cropping is done with respect to the reference or primary core point after the alignment process. The ratio of the top portion to the bottom portion is around 1 : 3 as the lower portion contains more information that is useful for classification. Fig. 6.5(c) shows a sample fingerprint image obtained after the cropping process.

Stage 5: Modeling of the orientation images

The constrained nonlinear orientation model is constructed based on the singular points that appear in the normalized image. The first order phase portrait, used to describe the local orientation around singular points is used to constrain the high order phase portrait which is used to describe the overall orientation of the image.

Here the high order phase portrait Eqn. 6.1 is rewritten in matrix forms as shown in Eqn. 6.5:

For each singular point present in the normalized orientation image, the coefficients of the first order phase portrait $\{m_i, n_i, c_i, d_i\}$ are extracted using the smallest absolute value of Eqn. 3.19. If the primary core is taken as the origin, a_{00} and b_{00}

are all set to 0 to ensure the primary point is a singular point.

$$\cos(2\theta) = (1 \ x \ \cdots \ x^n)PR \begin{pmatrix} 1 \\ y \\ \vdots \\ y^n \end{pmatrix} \quad (6.5a)$$

$$\sin(2\theta) = (1 \ x \ \cdots \ x^n)PI \begin{pmatrix} 1 \\ y \\ \vdots \\ y^n \end{pmatrix} \quad (6.5b)$$

where

$$PR = \begin{pmatrix} a_{00} & a_{01} & a_{02} & \cdots \\ a_{10} & a_{11} & & \\ a_{20} & & & \\ \vdots & & & \ddots \end{pmatrix} \quad (6.6a)$$

$$PI = \begin{pmatrix} b_{00} & b_{01} & b_{02} & \cdots \\ b_{10} & b_{11} & & \\ b_{20} & & & \\ \vdots & & & \ddots \end{pmatrix} \quad (6.6b)$$

Consequently, $3 * (K - 1)$ constraints are formed as shown in Eqn. 6.7:

$$\left\{ \begin{array}{l} \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} (-x)_s^{i-j} (-y)_s^j = 0 \\ \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} (i-j) (-y)_s^j = m_k \\ \sum_{i=0}^n \sum_{j=0}^i a_{(i-j)j} j (-x)_s^{i-j} = n_k \\ \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} (-x)_s^{i-j} (-y)_s^j = 0 \\ \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} (i-j) (-y)_s^j = c_k \\ \sum_{i=0}^n \sum_{j=0}^i b_{(i-j)j} j (-x)_s^{i-j} = d_k \end{array} \right. \quad (6.7)$$

where $k = 1 : K - 1$, K is the number of singular points present in the normalized orientation image. With these constraints, the upper triangle coefficient matrices PR and PI can be obtained using the least square algorithm. An example of the reconstructed orientation is shown in Fig. 6.5(d). All the orientation information are contained in these two coefficient matrices and subsequently used to perform classification.

Stage 6: Form the feature vectors

In this work, the feature vectors consist of two parts - the coefficients of constrained nonlinear phase portrait and the singularities information. Singularities information includes:

1. The number of singular points present in the original orientation image $\times$ the maximum magnitude of complex filters;
2. The angles between the primary core point and the delta points, if there is no delta, the corresponding angle will be set to 2π ;
3. The distance between the primary core point and the delta points, if there is no delta, the corresponding angle will be set to -1 .

Figure 6.6: Fingerprint type left loop, tented arch, right loop (from left to right). The solid lines show the orientation of the core points and dashed lines show the connection between core point and the delta point. The angle between these two lines is used to separate these three types of fingerprints.

The reason for using the first information instead of the number of singular points is that magnitudes of complex symmetrical filters provide a continuous way to measure the strength of singular points which is a more robust information compared to the number of singular points. The second information provides the relative difference in the core and delta points that cannot be determined from the orientation model (which provides only the orientation information). An illustration is shown In Fig. 6.6.

Stage 7: Perform fingerprint classification

Here a classifier is used to perform the fingerprint classification. Several classifiers are evaluated.

Neural Networks(NN) as the Classifier

We choose a simple back propagation (BP) feed-forward neural network with a single hidden layer as our classifier. Thus the classifier comprises three layers with the weights connecting between each neural in the different layers. The input layer consists of the exact number of input nodes corresponding to the number of feature vectors as obtained in stage 6. For the hidden layer, the network pruning method

is used to determine the optimal number of nodes which produces the lowest error rate. The output layer of the network consists of 5 neurons, which correspond to the number of fingerprint classes. A winner-take-all approach is used to determine the type of input fingerprint.

The training of the BP neural network is to iteratively change the weight by an amount proportional to the difference between the desired output and the actual output until it satisfies the predefined stopping criteria.

Support Vector Machine(SVM) As The Classifier

Unlike the learning strategy of neural networks which fine-tunes the weights of network by minimizing the distance between the desired outputs and the actual targets, Support Vector Machine finds the optimal hyperplane within the classes by maximizing the distance between the nearest points of the training data. The points near the hyperplane are named “Support Vectors”.

Without loss of generality, we explain the SVM by 2-class problem.

Given a training data $(x_i, y_i), i = 1, \dots, l$ where $x_i \in \mathbb{R}^n$, $y \in \{1, -1\}$ and l is the number of data. Define a kernel function by $K(\mathbf{x}, \mathbf{t}) = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{t})$. Then the decision function of SVM is written as:

$$f(\mathbf{x}) = \sum_{i=1}^l \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) \quad (6.8)$$

where the coefficients α_i could be found by the following constraints:

$$\begin{aligned} \min_{\alpha, b, \xi} \quad & \frac{1}{2} \alpha^T \alpha + C \sum_{i=1}^l \xi_i \\ \text{subject to} \quad & y_i(\alpha^T \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \quad \xi \geq 0 \end{aligned} \quad (6.9)$$

where $C > 0$ is the penalty parameter of the error term.

As be seen above, the most important step to implement the classification by SVM is to find a suitable kernels. Here we list four basic kernels

- Linear: $K(x_i, x_j) = x_i^T x_j$;
- Polynomial: $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \quad \gamma > 0$;
- Radial basis function (RBF): $K(x_i, x_j) = \exp(\gamma \|x_i - x_j\|^2), \quad \gamma > 0$;
- Sigmoid: $K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$;

where γ, r, d are kernel parameters.

The multiple classes problem can be solved by applying *one-vs-all* strategy. For more details, please refer to the Ref. [98].

Reduced Multivariate Polynomial (RM) Model As The Classifier

The RM model is first proposed for biometric decision fusion where the decision fusion problem is treated as a classifier fusion problem [99]. RM model is derived from the general multivariate polynomial (MP) model (as Eqn. 6.10 shows), which provide an effective way to describe complex nonlinear input-output relationships.

$$g(\alpha, x) = \sum_i^K \alpha_i x_1^{n1} x_2^{n2} \cdots x_l^{nl} \quad (6.10)$$

where the summation is taken over all non-negative integers $n1, n2, \dots, nl$ for which $n1 + n2 + \dots + nl \leq r$ with r being the order of approximation. $\alpha = [\alpha_1, \dots, \alpha_K]^T$ is the parameter vector to be estimated and x denotes the regressor vector $[x_1, \dots, x_l]^T$ containing l inputs. K is the total number of terms in $g(\alpha, x)$.

MP model suffers from the exponential increase in the amount of coefficients as the number of inputs and the order increases. When the order of MP model is r , the

number of coefficients reaches l^r , here l is the dimension of the input data. Thus the MP needs huge training data to ensure the coefficients are determined in a stable manner.

To overcome the shortcomings of MP model, the RM model can be written as the following expression:

$$\begin{aligned} \hat{f}_{RM}(\alpha, x) = & \alpha_0 + \sum_{j=1}^l \alpha_j x_j + \sum_{j=1}^r \alpha_{l+j} (x_1 + x_2 + \cdots + x_l)^j + \\ & \sum_{j=2}^r (\alpha_j^T x) (x_1 + x_2 + \cdots + x_l)^{j-1} \quad l, r \geq 2 \end{aligned} \quad (6.11)$$

where the number of terms is given by $K = 1 + r(l + 1)$.

To include more individual high-order terms, the following (RM) can be written in Eqn. 6.12:

$$\begin{aligned} \hat{f}_{RM}(\alpha, x) = & \alpha_0 + \sum_{j=1}^r \sum_{k=1}^l \alpha_{kj} x_j^k + \sum_{j=1}^r \alpha_{l+j} (x_1 + x_2 + \cdots + x_l)^j + \\ & \sum_{j=2}^r (\alpha_j^T x) (x_1 + x_2 + \cdots + x_l)^{j-1} \quad l, r \geq 2 \end{aligned} \quad (6.12)$$

The number of terms in this model can be expressed as $K = 1 + r + l(2r - 1)$. It is noted that the second Eqn. has $(rl - l)$ number of terms more than the first.

6.5 Experimental Results

6.5.1 Dataset

The NIST Special Database 4 (DB4), which contains 2000 pairs of fingerprint images (two instances, f and s, for each entity), is used to evaluate the proposed fingerprint

classification algorithm. Each image in DB4 has 8-bit grey-level with a size of 512×480 pixels. The five classes: Arch (A), Left Loop (L), Right Loop (R), Tented Arch (T) and Whorl (W), are uniformly distributed in the database. The images are numbered from $f0001$ to $f2000$ and from $s0001$ to $s2000$. In DB4, there exist about 17% ambiguous fingerprints that have additional ‘secondary’ class assigned to them. Therefore, in our experiment, output that corresponds to either the primary class or secondary class is taken as a correct classification. This consideration is also adopted by other researchers [1, 4, 51, 71, 72]. Fig. 6.7 below shows a fingerprint image that has 2 classes, its primary class is Tented Arch (T), and its secondary class is Left Loop (L).

Figure 6.7: An example of an ambiguous fingerprint class

6.5.2 Performance Measurement

The performance of the fingerprint classification system is measured in terms of error rate or accuracy. The error rate is computed as the ratio between the number of misclassified fingerprints and the total number of samples in the test set. The

accuracy is thus the percentage of correctly classified fingerprints.

$$errorrate = \frac{\text{number of misclassified fingerprints}}{\text{total number of fingerprints}} \times 100\% \quad (6.13)$$

$$accuracy = 100\% - errorrate \quad (6.14)$$

6.5.3 Coefficients Used for Classification

The order of the constrained nonlinear orientation model is a tradeoff between the accuracy of reconstructed orientation and the conciseness of the model. Based on our extensive experiment, the order of constrained nonlinear phase portrait model is set to 6 as it can model the orientation well while only requiring a total of 54 coefficients.

Table 6.3: Classification results with different amounts of coefficients

	Up to order 4	Up to order 5	full 6 order
classification accuracy	87.4%	87.6%	87.9%

In these 54 coefficients, the first order coefficients $\{a_{01}, a_{10}, b_{01}, b_{10}\}$ exhibit the behavior of the primary core point while the higher order coefficients act as the correction terms. With higher order coefficients, the correction terms will affect larger areas of the fingerprint. Fig. 6.8 shows the examples of reconstructed orientation with different number of coefficients as the correction terms. Fig. 6.8(f) shows the orientation obtained when all the coefficients are used to reconstruct the orientation while Fig. 6.8(a)– 6.8(e) show the orientation obtained when only coefficients up to order 1, 2, 3, 4 and 5 respectively of the 6-order model are used to reconstruct the orientation. As expected, Fig. 6.8 shows that as the order of the coefficient used increases, the size of the region with correctly reconstructed orientation increases. Using coefficients up to order 6, the orientation of the entire cropped image can be

correctly reconstructed. Since the important pattern of a fingerprint is mainly characterized by the orientation near the primary core point region, we hypothesized that using the first k order coefficients, here $4 \leq k \leq 6$, is sufficient to perform fingerprint classification with reasonable accuracy. To validate this hypothesis, we feed the coefficients up to order 4, 5 and 6 respectively from the 6-order constrained nonlinear model as the input to the classifier respectively. Table 6.3 shows the classification accuracy achieved.

(a) reconstructed orientation using the coefficients of the first 1 order

(b) reconstructed orientation using the coefficients of the first 2 order

(c) reconstructed orientation using the coefficients of the first 3 order

(d) reconstructed orientation using the coefficients of the first 4 order

(e) reconstructed orientation using the coefficients of the first 5 order

(f) reconstructed orientations using the full coefficients

Figure 6.8: Reconstructed orientation using the different order coefficients

From Table 6.3, the accuracy of the classification achieved using the number of

coefficient up to order 4 and with the full 6 order is quite similar. However, with order 4, the number of the input nodes of the classifier is reduced significantly, thus simplifying the structure of the classifier. Therefore, we have opted to use only coefficients up to order 4, which results in only 28 input nodes.

6.5.4 Classification Results Using Orientation Model

In our experiment, all images are being used regardless of its quality. The images in the database are divided into the training set and the testing set. The training set includes 2000 images of the first 1000 pairs for both “f” and “s” instances. The rest of images are used as the testing set.

The parameters of the classifier are determined using cross validation technique whereby in the training set “ $f0001 - f1000$ ” and “ $s0001 - s1000$ ”, the data is divided into several subset, one of the subset will be used as training data, and the performance is measured using the other subsets. The final parameters are the parameters of the classifier which produces the best performance.

In this section, the experimental results using three classifiers will be reported. The number of input vectors is 28 from the first 4 order coefficients of the model) + 6 from the singularities information according to the analysis in Sec.6.5.3.

6.5.4.1 NN as the Classifier

The structure of BP neural network is determined by the following steps.

We started off with 200 hidden nodes and pruned the number of hidden nodes to 10. In each turn, the average error rates for 5 times cross validations were used as a basis to obtain the optimal network configuration.

From the graph (see Fig. 6.9), it can be seen that the error rates for the various network configuration is in the range of 7% to 9%. This implies that the neural networks are capable of performing reasonably good fingerprint classification. Also the optimal number of hidden nodes is about 100 nodes.

Figure 6.9: Graph of error rate vs. number of hidden nodes

Using the optimal network configuration, we simulated cross validation again for 30 times and obtained the best parameters for neural network classifier using which, we can get the lowest error rate for the training set. The average error rate of 50 times cross validations is 7.59% with a standard deviation 0.0981. And the lowest error rate of the training set is 7.00%.

Using this best neural network obtained from the training set, we get an accuracy rate of 92.1% for the whole testing set for 5-class problem. Considering that the tented arch and the plain arch are not very common in nature, these 2 classes can actually be combined together to give a four class classification. Combining the 2 classes, our method yields an accuracy rate of 94.05%.

To obtain a more detailed analysis of the neural network classifier, the confusion

Table 6.4: Confusion matrix with NN as the classifier

True Class	Hypothesized Class					Total
	L	R	W	T	A	
L	363	0	1	24	18	406
R	2	359	0	27	12	400
W	14	13	371	1	0	399
T	3	4	0	316	22	345
A	0	0	0	17	433	450
Grand Total						2000

L=Left Loop, R=Right Loop, W=Whorl, T=Tented Arch, A=Plain Arch

matrix is plotted in Table 6.4. This matrix has a row for each true class and a column for each hypothesized class.

6.5.4.2 SVM as the Classifier

We employ the publicly available OSU SVM Classifier Matlab Toolbox version 3.00 [100]. We set the kernel as the RBF function. The kernel parameters are selected using the cross-validation and grid-search technique. Firstly we separate the training data into n folds, then one subset is tested for each parameter pairs $C = 2^{-5}, 2^{-3}, \dots, 2^{15}, \gamma = 2^{-15}, 2^{-13}, \dots, 2^3$ using the classifier trained on the remaining $v - 1$ subset until the best cross-validation accuracy is obtained.

In our experiments, the best parameters for g and c are set to $c = 2^4; g = 2^{-3}$. The accuracy rate is 94.35% for 5 classes problem. For 4 classes problem, the accuracy rate is 95.56%.

The confusion matrix is shown in table 6.5.

6.5.4.3 RM Model as the Classifier

The computational details of RM model can be found in [101]. There are two parameters to be determined in RM model. Here we set the order of the model from 1

Table 6.5: Confusion matrix with SVM as the classifier

True Class	Hypothesized Class					Total
	L	R	W	T	A	
L	396	0	2	13	18	429
R	4	412	1	4	14	435
W	8	11	380	0	0	399
T	4	5	0	237	24	270
A	3	2	0	0	462	467
Grand Total						2000

L=Left Loop, R=Right Loop, W=Whorl, T=Tented Arch, A=Plain Arch

to 9 and the regularization parameter from 10^{-4} to 10^{-1} . For the five output class of Arch (A), Left Loop (L), Right Loop (R), Tented Arch (T) and Whorl (W), the simulation results are listed in Table 6.6.

From Table 6.6, the RM model with order 2 produces the lowest classification error rate of 9.1%. To give a more detailed analysis of the classification results using RM as the classifier, the confusion matrix is shown in Table 6.7.

The parameters with the best RM result of cross validation is applied to the testing set, the accuracy rate reaches 91.85% for 5-class problem and 93.2% for 4-class problem. And the corresponding confusion matrix is shown in Table 6.7.

The results show that the SVM classifier gave the best accuracy rate among the 3 classifiers used. As a comparison, we include another 4 tables showing the confusion matrix obtained using the following approaches

- "fingercod" as feature and neural network as classifier;
- "fingercod" as feature and support vector machine as classifier;
- singularities and pseudo-ridge information as features and the classifier is rule based;
- Multi- KL transform of the orientation image as feature and the combinational

Table 6.6: Classification results by RM

Model	Regularization Parameters	Error rate (%)
1	10^{-4}	9.5
	10^{-3}	9.5
	10^{-2}	9.4
	10^{-1}	9.4
2	10^{-4}	9.2
	10^{-3}	9.2
	10^{-2}	9.1
	10^{-1}	9.1
3	10^{-4}	9.6
	10^{-3}	9.6
	10^{-2}	9.6
	10^{-1}	9.3
4	10^{-4}	10.7
	10^{-3}	10.7
	10^{-2}	10
	10^{-1}	9.3
5	10^{-4}	12
	10^{-3}	11.9
	10^{-2}	10.3
	10^{-1}	9.7
6	10^{-4}	15.7
	10^{-3}	14.7
	10^{-2}	12.7
	10^{-1}	10.8
7	10^{-4}	18.4
	10^{-3}	16.4
	10^{-2}	13.5
	10^{-1}	10.7
8	10^{-4}	19.7
	10^{-3}	17.7
	10^{-2}	14.9
	10^{-1}	12.1
9	10^{-4}	24.1
	10^{-3}	19.8
	10^{-2}	17.7
	10^{-1}	13.8

Table 6.7: Confusion matrix with RM as the classifier

True Class	Hypothesized Class					Total
	L	R	W	T	A	
L	361	3	1	24	25	414
R	4	368	0	21	17	410
W	16	16	367	0	0	399
T	3	5	0	283	22	313
A	0	1	0	5	458	464
Grand Total						2000

L=Left Loop, R=Right Loop, W=Whorl, T=Tented Arch, A=Plain Arch

classifiers are used to perform the classification.

The fingercode methods with neural network and SVM as classifiers respectively give the similar accuracy rates of 90%. With only one feature, the 90% accuracy rate obtained is indeed a good classification performance. In Zhang's work, the pseudo-ridge information is used to complement the singular points. The tracing of the pseudo-ridge strongly depends on the orientation. If only the local orientations is considered, it is difficult to correctly trace the ridge in noisy images. The Multi-KL transform based method give the best reported result as far as we know. This is because in this method, the orientation image is divided into several subspaces. In each subspace, KL transform is applied. Thus more details are extracted for classification. The combination of two different classifiers also help to improve the classification accuracy rate. This is the way to improve our method in future.

More classification results reported by the other researchers are listed in Table 6.12 together with the result of our proposed approach for comparison purpose. The table shows that our results are competitive on the NIST database 4 for the 4 and 5 classes problem, where our approach using the SVM classifier is ranked second after the approach by Cappelli *et. al.* (2003) [3].

Table 6.8: Confusion matrix with NN as the classifier by the Jain's fingercode method [1]

True Class	Hypothesized Class					Total
	L	R	W	T	A	
L	364	0	6	7	6	383
R	1	372	3	17	8	401
W	8	16	366	1	4	395
T	14	6	0	261	55	336
A	3	1	2	39	405	450
Grand Total						1965*

L=Left Loop, R=Right Loop, W=Whorl, T=Tented Arch, A=Plain Arch

*: rejection criteria used here cause the amount of images less than 2000

Table 6.9: Confusion matrix with SVM as the classifier by both the fingercode and RNN-extracted features [2]

True Class	Hypothesized Class					Total
	L	R	W	T	A	
L	357	1	6	13	2	379
R	0	354	5	29	7	392
W	8	18	366	2	0	394
T	12	8	1	294	48	363
A	2	2	0	33	396	433
Grand Total						1965*

L=Left Loop, R=Right Loop, W=Whorl, T=Tented Arch, A=Plain Arch

*: rejection criteria used here cause the amount of images less than 2000

Table 6.10: Confusion matrix by the Cappelli's method [3]

True Class	Hypothesized Class					Total
	L	R	W	T	A	
L	373	2	4	3	4	386
R	1	381	9	4	3	398
W	4	7	391	0	0	402
T	3	7	5	373	6	394
A	7	10	1	15	387	420
Grand Total						2000*

L=Left Loop, R=Right Loop, W=Whorl, T=Tented Arch, A=Plain Arch

6.6 Summary

In this paper, orientation image and singularity information are combined as the input features for fingerprint classification. The orientation information is obtained

Table 6.11: Confusion matrix by the Zhang's method [4]

True Class	Hypothesized Class					Total
	L	R	W	T	A	
L	739	5	25	0	44	813
R	14	721	24	3	51	813
W	27	26	727	5	3	788
T	31	12	18	278	293	632
A	7	6	4	11	908	936
Grand Total						3982*

L=Left Loop, R=Right Loop, W=Whorl, T=Tented Arch, A=Plain Arch

*: 18 images belong to unknown type

from the orientation model computed using the constrained nonlinear phase portrait model approach while the singularity information is obtained from the response of the complex symmetrical filters. As the computation of the orientation model and singular points are interdependent, an interactive validation approach is suggested whereby potential candidate singular points are detected and then verified using the difference in the orientation model at the singular point region with the original orientation. This improves both the accuracy of the singular point detection and the orientation model. The combined orientation and singularity features are used to classify fingerprints using an SVM classifier. The classification accuracy on NIST database 4 is shown to be good, achieving a rate of 94.35% for 5-class problem and 95.56% for 4-class problem.

In this study, all images are used to perform fingerprint classification regardless of their quality. However, some of the orientation image cannot be reconstructed well due to the noise. Thus the usage of such features will affect the parameter selection during the training and decrease the classification accuracy during the testing. In future, two-stage training will be considered with the initial stage training not considering the poor quality fingerprint images. Furthermore, since more subtle orientation image description in the subclass spaces could result in better classification

accuracy, using the constrained nonlinear phase portrait model to discriminate such subclass feature deserves further investigation.

Table 6.12: Literature reviews of fingerprint classification

Algorithms	Features	Classifier	DB4 Results		Comments
			5 classes	4 classes	
Kawagoe & Tojo (1984) [49]	SP & RI	Rule-Based	-	-	94 Images and 7 Classes
Wilson <i>et al.</i> (1994) [78]	OI	NN	90.2%	-	64 features and 10% rejection
Candela <i>et al.</i> (1995) [68]	OI & RI	NN & Rule-based	-	88.6%	PCA features from OI
Karu and Jain (1996) [51]	SP	Rule-Based	85.4%	91.1%	-
Fitz & Green(1996) [92]	Hexagonal FT	k-NN	-	-	40 Images and 3 classes
Halici & Ongun(1996) [69]	OI	SOM	-	-	self-defined classes By SOM
Chong <i>et al.</i> (1997) [79]	RI	Rule-based	-	-	86 Images, 5 classes(A,L,R,W,D)
Jain <i>et al.</i> (1999) [1]	GF	K-NN & NN	90%	95.8%	1.8% rejects. 192 features.
Cappelli <i>et al.</i> (1999) [91]	OI	the Minimum Cost	87.1%	-	MKL features from OI
Senior (2001) [80]	RI	HMM & Decision Trees	-	94.9%	Nature distribution
Pattichis <i>et al.</i> (2001) [102]	OI	NN	-	-	OI gotten from FM element
Jain & Minut (2002) [81]	OI	Rule-based	-	91.25%	kernels defined from OI
Cappelli <i>et al.</i> (2002) [90]	OI	k-NN & Minimum Cost	92.2%	-	PCA features from OI
Chang & Fan (2002) [82]	RI	Rule-based	94.8%	-	5.1% rejects; 93.4% for 7 classes
Shah & Sastry (2003) [70]	RI	SVM, NN, k-NN	-	-	DB14; 96 and 128 features
Cappelli <i>et al.</i> (2003) [3]	OI	Minimum cost & quadratic DF	95.2%	96.3%	-
Yao <i>et al.</i> (2003) [72]	OI & GF	SVM, RNN	90.0%	94.7%	1.8% rejects;
Nyongesa <i>et al.</i> (2004) [2]	SP	RBF, MLP & FNN	93.75%	-	-
Zhang & Yan (2004) [4]	SP & RI	Rule-based	84.3%	92.7%	-
Dass & Jain (2004) [83]	RI	Rule-based	-	94.4%	-
Park & Park (2005) [93]	FT	Nonlinear-DA	90.7%	94.0%	-
Neuhaus & Bunke (2005) [85]	OI variance	NN	86.01%	-	-
Our method	OI & SP	SVM	94.35%	95.56%	Modeling of OI and singularities as features

SP= Singular Point; RI = Ridge Information; FT = Fourier Transform; HMM = Hidden Markov Model; k-NN = k-Nearest Neighbor; RM = Reduced Polynomial Model
OI = Orientation Image; DF=Discriminant Function; NN= Neural Networks;GF = Gabor Filters; SOM = self-Organizing Mapping; SVM=Supported Vector Machine;

Chapter 7

Conclusions And Future Works

7.1 Conclusion

This thesis focuses on fingerprint orientation model. A unified orientation model is proposed to precisely compute the fingerprint orientation. The proposed approach comprises orientation prediction and constrained nonlinear phase portrait modeling. For the prediction model, a first order phase portrait model is used to reconstruct the orientation images near singular points. The different description of singular points are then combined in piecewise manner to reconstruct the whole orientation model. To cater for cases where some of singular points are missing in the fingerprint image, a search algorithm is proposed to find the possible missing singular points. We showed that the efficiency of the search algorithm can be increased by predefining 37 regions in the orientation image where singular points can be located. The resultant prediction model is able to predict the orientation image at the corrupted fingerprint region with unreliable or missing information.

Although the prediction model has a linear order with strong regularity for good

generalization around a singular region, it does not have sufficient degree of freedom to estimate the global orientation of a fingerprint which contains nonlinear orientation. To overcome this limitation, we use the nonlinear phase portrait model which has higher order. We constrain the coefficients of the nonlinear phase portrait model with the first order phase portrait model so that the overall model is able to accurately reconstruct the orientation of the overall fingerprint image as well as the orientation near the singular point regions. The resultant formulation gave a unified form for accurate fingerprint orientation modeling.

The proposed strategy of combining the orientation prediction and the constrained nonlinear phase portrait modeling brings the accuracy of the orientation model to a new level. The prediction model replaces the missing or noisy orientation with predicted orientation and remains functioning despite missing singular points by providing a search for these missing singular points. Subsequently, the corrected orientation image is used by the constrained nonlinear phase portrait to accurately model the orientation of the entire fingerprint image and is robust against noisy or missing orientation information. Experimental results show that the proposed approach is able to correctly model the fingerprint orientation for all classes and as well as the noisy fingerprint images in which there may exist a large noise patch.

Since the fingerprint orientation pattern is crucial for fingerprint classification, we postulate that the fingerprint orientation model itself can be used to perform fingerprint classification. This thesis proposed a fingerprint classification algorithm based on the constrained nonlinear phase portrait orientation model. As the model is affected by variation in the position and orientation of the fingerprint, we employ the symmetrical complex filters to normalize the fingerprint image, as well as to detect singular points. The orientation model of the normalized fingerprint image is then computed and the coefficients up to order 4 (or 28 parameters) are extracted to form a

feature vector. Since the orientation model lacks the singular point information, these information obtained from the detected singular points are then added to the feature vectors. An SVM classifier is then used to train the feature vector for fingerprint classification. Experimental results on the NIST database 4 show that the proposed algorithm has good classification accuracy, which ranked 2nd among the best known results from the state-of-the-art.

7.2 Future Work

We conclude this thesis by suggesting some possible future works that complement the research works presented in this thesis:

1. Improve the performance of orientation modeling as follows:
 - (a) Develop a reliable image quality evaluation approach. In this thesis, I used the orientation coherence value as the fingerprint quality index. Coherence value is a coarse estimation of the image quality. It will be affected by several factors such as the existence of minutiae or scar. It also cannot distinguish the high curvature orientation around singular points from noise. Nevertheless, our experimental results show that using the coherence index as the quality index produces acceptable results for almost all the images. A better approach to estimate the quality of the image will be able to improve the results further. In addition, analyzing how the reliability of the image quality estimation affects the performance of the orientation model is also a useful future work.
 - (b) Develop an orientation modeling approach that is not dependent on singular-point. Dependency on the singular point detection is the main limitation

for all the model based methods till now, including ours. There are two possible approaches to solve this: (1) improve accuracy of singular points detection; (2) develop a new method which is not dependent on the singular point.

2. Improve the performance comparison among the various proposed orientation modeling approaches. It is difficult to obtain a ground truth in noisy images. Currently, we used the orientation estimated using Gabor filter as the basis for comparison. The same evaluation method can be found in Gu's work [60]. The ground truth may not be the best one, however, the orientation obtained from the Gabor filter would be a good indicator and thus a quantitative comparison among the different orientation model could be established. Such quantitative comparison more likely emphasizes the amount of differences among the various proposed orientation model instead of the absolute number. Therefore, providing comparison which includes qualitative evaluation based on the error and its relation to the theoretical analysis of the proposed approach will enhance the correctness of the comparison. Alternatively, a better quantitative comparison can be done through the following two possible ways:

- (a) Using synthetic fingerprint images; Cappelli et al. [9] have proposed a method to synthesize fingerprint images. The advantage of using synthetic fingerprint images is that the ground truth of the orientation is always known. (In their method, the orientation image is first produced; the noise will be added into the ideal orientation image later.) Unfortunately we are not able to get the ground truth orientation before the noise is added into the images from the released version for our experiments. Developing a new synthetic system is out of the scope of this work.

- (b) Using the fingerprint verification system to test the performance of the orientation modeling. This is an indirect way to examine the performance of the orientation modeling. For a given fingerprint verification system, two sets of minutiae feature are formed. The difference between this two feature sets is only on the orientation information; one is from the proposed method, the other from the other method to be compared. However, this would require the development of a fingerprint verification system which is out of the scope of this thesis. We are not able to use a commercial off-the-shelf fingerprint verification system as the system does not allow us to modify the orientation block and we cannot modify the fingerprint image. Nevertheless, we believe that once we are able to integrate it into a fingerprint verification system, the correct orientation provided by our proposed approach will improve the matching performance. This would be another future work.
3. Continuous classification. In this thesis, only exclusive classification of fingerprints is discussed. Unfortunately, the distribution of the fingerprints among the 5 classes is not uniform. Hence, some researchers proposed “continuous classification” [20, 103, 104] to obtain better classification scheme for very large databases. In continuous classification, every fingerprint will be assigned with a score or a feature vector. Given a query image, the similar score or feature vector is computed and compared. The first N most likely fingerprints are taken as belonging to the same class. Similar approach can be investigated using the coefficients of the normalized constrained nonlinear phase portrait model. One of the challenges is to determine the effect of small variations in the position and orientation of the fingerprint image present even in the normalized fingerprint images and determining a way to normalize these coefficients.

4. Synthesis of the fingerprint template. In [37] Yau *et al.* showed that the template of several small fingerprint images can be merged into one that is equivalent to the template derived from a large image. The basic assumption made is that there are sufficient minutia points present in the overlapping region among the different small images to establish the template alignment. However, for cases where the overlap region is insufficient or non-existence such as in the case of sensor array, then means to reconstruct the overall fingerprint image or template from the array of images is needed. For such problem, the missing orientation can be predicted and used to guide the generation of the missing ridge lines or to align these templates based on the approach proposed in this thesis together with the ridge generation approach using the AM-FM framework proposed by Acton *et al.* [105].

Appendix A

Reconstruction Of The Delta Pattern From Two Arbitrary Asymptotes

In a cartesian coordinates system (x', y') , we suppose that the delta pattern can be described by the first order phase portrait model as Eqn.A.1:

$$\frac{dy'}{dx'} = A \frac{x'}{y'}; \tag{A.1}$$

where $A > 0$ is a coefficient. (x', y') are the coordinates. Fig. A.1 show the examples of the delta type orientation patterns with $A = 1, 3, 0.3$ respectively.

Consider a function $F' = \frac{1}{2}(A * x'^2 - y'^2) + C = 0$, where C is a constant number. With $C = -\infty \rightarrow +\infty$, the function F' consist of a family of hyperbolas. The set of the orientation of an arbitrary point (x', y') represent the delta pattern.

Take the derivative on both sides of the function F' , we have the following func-

Figure A.1: Examples of the delta pattern with different A

tion:

$$\frac{\partial F'}{\partial x'} + \frac{\partial F'}{\partial y'} \frac{\partial y'}{\partial x'} = 0 \quad (\text{A.2})$$

From Eqn.A.2, we can obtain

$$\frac{dy'}{dx'} = -\frac{\partial F'}{\partial x'} / \frac{\partial F'}{\partial y'} = -\frac{Ax'}{-y'} = A \frac{x'}{y'} \quad (\text{A.3})$$

Compared Eqn. A.1 with Eqn.A.3, we find that the function F' is one of the solutions of Eqn.A.1. The two asymptotes of the function F' can be represented by the following Eqn.A.4:

$$y' = \pm \sqrt{A} x' \quad (\text{A.4})$$

All these three functions A.1, A.2 or A.3 share the same and unique coefficient A , thus if we know any of the Eqns. A.1, A.2 or A.3, we can reconstruct the orientation pattern of the function F' using the coefficient A .

If we know two arbitrary asymptotes of a delta type orientation, they can be obtained by rotating the Eqns. A.4 with a certain angle θ , thus its orientation pattern can be reconstructed by rotating the function F' with the same angle θ .

In the following section, we give the details to reconstruct the delta type of orientation pattern by two arbitrary asymptotes.

Suppose we have two arbitrary asymptotes in the new cartesian coordinates system (x, y) , which is obtained by rotating the (x', y') system with a certain angle θ as the Fig. A.2 shown.

Figure A.2: Explanation of the reconstruction using the two asymptotes

$$y - k_1x = 0 \quad (\text{A.5a})$$

$$y - k_2x = 0 \quad (\text{A.5b})$$

Let $\theta_1 = \arctan^{-1}(k_1)$ and $\theta_2 = \arctan^{-1}(k_2)$ represent the slopes of these two asymptotes respectively.

Rotate these two asymptotes with the angle $\theta = -(\theta_1 + \theta_2)/2$ counterclockwise, we have new two asymptotes in the new coordinate system (x', y') :

$$y' \pm \tan\left(\frac{\theta_1 - \theta_2}{2}\right)x' = 0 \quad (\text{A.6})$$

From these two asymptotes, we can get a function $F' = \tan^2\left(\frac{\theta_1 - \theta_2}{2}\right)x'^2 - y'^2 = 0$,

the orientation field of function F' show a delta behavior. Using the first order phase portrait approach, its orientation field can be represented as the following Eqns.:

$$\frac{dy'}{dx'} = K \frac{x}{y} \quad (\text{A.7})$$

where $K = \tan^2(\frac{\theta_1 - \theta_2}{2})$.

Now we rotate the function F' with an angle $\theta = (\theta_1 + \theta_2)/2$, we have a new function F in the x, y coordinate system

$$F = K(x \cos(\theta) + y \sin(\theta))^2 - (-x \sin(\theta) + y \cos(\theta))^2 \quad (\text{A.8})$$

Its orientation field can be represented by the first order phase portrait model:

$$\frac{dy}{dx} = -\frac{\partial F}{\partial x} / \frac{\partial F}{\partial y} = -\frac{(K \cos^2 \theta - \sin^2 \theta)x + (K \cos \theta \sin \theta + \cos \theta \sin \theta)y}{(K \cos \theta \sin \theta + \sin \theta \cos \theta)x + (K \sin^2 \theta - \cos^2 \theta)y} \quad (\text{A.9})$$

Proved.

Author's Publications

Book Chapter

1. **Jun Li**, 'Orientation modeling in noisy fingerprint images', *in a book of Biometrics Recognition 2007* (accepted) (with Dr. Wei Yun Yau)

Journal Publications

2. **Jun Li**, Wei Yun Yau, Han Wang, 'Constrained Nonlinear Models of Fingerprint Orientations With Prediction', *Pattern Recognition*, 39/1 pp. 102-114 (doi: 10.1016/j.patcog.2005.08.010)
3. **Jun Li**, Wei Yun Yau, Han Wang, 'Combining Singular Points and Orientation Image Information for Fingerprint Classification', *Pattern Recognition*, 2007 (Accepted)
4. S.L. Goh, **Jun Li**, Wei Yun Yau, K.A. Toh and D. Srinivasan, 'Fingerprint Classification Based on Orientation Model', *Multimedia Cyberspace Journal Special Issue on Pattern Recognition in Biometrics and Bioinformatics*, vol. 9, Vol. 3 No. 3, 2005 pp 39 - 46

Conference Publications

5. **Jun Li**, Wei Yun Yau, Han Wang, 'Fingerprint Indexing Based On Symmetrical Measurement' (Oral Presentation), *the 18th International Conf. on*

-
- Pattern Recognition (ICPR'06)*, Vol. 1, Aug. 20-24, 2006, pp 1038-1041 (doi: 10.1109/ICPR.2006.568)
6. **Jun Li**, Wei Yun Yau, Han Wang, 'Singular Points Detection Using Interactive Mechanism in Fingerprint Images' (Oral Presentation), *ICARVS 06*, Dec. 2006
 7. **Jun Li**, Wei Yun Yau, 'Prediction of Fingerprint Orientation', *the 17th International Conf. on Pattern Recognition (ICPR'04)*, Vol. 4, Aug. 23 - 26, 2004, Cambridge, UK, pp 436-439 (doi: 10.1109/ICPR.2004.1333796)
 8. Wei Yun Yau , **Jun Li**, Han Wang, 'Nonlinear Phase Portrait Model for Orientation Fields of Fingerprints', *In Proc. 8th International Conf. on Control, Automation, Robotics and Vision (ICARCV'2004)*, Dec. 2004, Kunming, China
 9. W.Y. Yau, S.L. Goh, **Jun Li**, K.A. Toh and D. Srinivasan, 'Using Unified Orientation Model for Fingerprint Classification', *DCDIS 4th International Conf. on Engineering Applications and Computational Algorithms*, July 27-29, 2005, Ontario, Canada

Bibliography

- [1] A. K. Jain, S. Prabhakar, and H. Lin. A multichannel approach to fingerprint classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4):348–359, 1999.
- [2] H. O. Nyongesa, S. Al-Khayatt, S. M. Mohamed, and M. Mahmoud. Fast robust fingerprint feature extraction and classification. *Journal of Intelligent and Robotic Systems*, 40(1):103 – 112, May 2004.
- [3] R. Cappelli, D. Maio, D. Maltoni, and L. Nanni. A two-stage fingerprint classification system. In *WBMA '03: Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications*, pages 95–99, 2003.
- [4] Q. Zhang and H. Yan. Fingerprint classification based on extraction and analysis of singularities and pseudo ridges. *Pattern Recognition*, 37(11):2233–2243, 2004.
- [5] D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar. *Handbook of Fingerprint Recognition*. Springer, 2002.
- [6] L. Hong, Y. Wan, and A.K. Jain. Fingerprint image enhancement: Algorithm and performance evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):777–789, August 1998.

-
- [7] B.G. Sherlock and D.M. Monro. A model for interpreting fingerprint topology. *Pattern Recognition*, 26(7):1047–1055, July 1993.
 - [8] P.R. Vizcaya and L.A. Gerhardt. A nonlinear orientation model for global description of fingerprints. *Pattern Recognition*, 29(7):1221–1231, 1996.
 - [9] R. Cappelli, A. Erol, D. Maio, and D. Maltoni. Synthetic fingerprint-image generation. In *International Conference on Pattern Recognition 2000*, pages Vol III: 471–474, 2000.
 - [10] M. A. Fischler and H. C. Wolf. Linear delineation. In *Conference on Computer Vision and Pattern Recognition*, pages 351–356, 1983.
 - [11] S. W. Zucker. Early orientation selection: tangent fields and the dimensionality of their support. In *Papers from the second workshop Vol. 13 on Human and Machine Vision II*, pages 335–364, 1986.
 - [12] M. Kass and A. Witkin. Analyzing oriented patterns. *Computer Vision Graphics and Image Processing*, 37(3):362–385, 1987.
 - [13] I. Cohen and I. Herlin. Non uniform multi-resolution method for optical flow and phase portrait models: environmental applications. *International Journal of Computer Vision*, 33:29–49, 1999.
 - [14] X. Jiang. A study of fingerprint image filtering. In *International Conference on Image Processing 2001*, pages III: 238–241, 2001.
 - [15] J. Yang, L. Liu, T. Jiang, and Y. Fan. A modified gabor filter design method for fingerprint image enhancement. *Pattern Recognition Letter*, 24(12):1805–1817, August 2003.

- [16] B. Rusyn, I. Prudyus, and V. Ostap. Fingerprint image enhancement algorithm. In *CAD Systems in Microelectronics, 2001. CADSM 2001. Proceedings of the 6th International Conference.*, pages 193–194, 2001.
- [17] B.G. Sherlock, D.M. Monro, and K. Millard. Algorithm for enhancing fingerprint images. *Electronics Letters*, 28(18):1720–1721, Aug. 1992.
- [18] C.T. Hsieh, E. Lai, and Y.C. Wang. An effective algorithm for fingerprint image enhancement based on wavelet transform. *Pattern Recognition*, 36(2):303–312, February 2003.
- [19] K. Ito, T. Aoki, and T. Higuchi. Design of a digital reaction-diffusion system for restoring blurred fingerprint images. In *Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on*, pages IV–77– IV–80 vol.4, 2003.
- [20] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni. Fingerprint classification by directional image partitioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):402–421, 1999.
- [21] D. Maio and D. Maltoni. A structural approach to fingerprint classification. In *Proceedings of the International Conference on Pattern Recognition (ICPR '96)*, volume III, pages 578–585, 1996.
- [22] S. Bernard, N. Boujemaa, D. Vitale, and C. Bricot. Fingerprint classification using kohonen topologic map. In *International Conference on Image Processing(ICIP01)*, pages III: 230–233, 2001.
- [23] A.R. Roddy and J.D. Stosz. Fingerprint features-statistical analysis and system performance estimates. *Proceedings of the IEEE*, 85(9):1390–1421, Sep. 1997.

-
- [24] G. Bebis, T. Deaconu, and M. Georgiopoulos. Fingerprint identification using delaunay triangulation. In *Information Intelligence and Systems 1999. Proceedings. International Conference on*, pages 452 – 459, 1999.
 - [25] Z. Chen and C.H. Kuo. A topology-based matching algorithm for fingerprint authentication. In *Proceedings. 25th Annual 1991 IEEE International Carnahan Conference on Security Technology*, 1991.
 - [26] D. Maio and D. Maltoni. Direct gray-scale minutiae detection in fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(1):27–40, January 1997.
 - [27] N. K. Ratha, K. Karu, S. Chen, and A. K. Jain. A real-time matching system for large fingerprint databases. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):799–813, 1996.
 - [28] S.H. Chang, F.H. Cheng, W.H. Hsu, and G.Z. Wu. Fast algorithm for point pattern-matching: Invariant to translations, rotations and scale changes. *Pattern Recognition*, 30(2):311–320, 1997.
 - [29] X. Jiang and W.-Y. Yau. Fingerprint minutiae matching based on the local and global structures. In *Proc. 15th International Conference on Pattern Recognition (ICPR 2000)*, pages Vol.2, pp.1042–1045, 2000.
 - [30] K.C. Fan, C.W. Liu, and Wang Y.K. A randomized approach with geometric constraints to fingerprint verification. *Pattern Recognition*, 33(11):1793–1803, 2000.
 - [31] R. Udupa, G. Garg, and P. Sharma. Fast and accurate fingerprint verification. In *the 3rd Audio- and Video-Based Biometric Person Authentication*, pages 192–197, 2001.

- [32] A.K. Jain, L. Hong, S. Pankanti, and R.M. Bolle. An identity-authentication system using fingerprints. *Processings of the IEEE*, 85(9):1365–1388, September 1997.
- [33] L. Hong and A.K. Jain. On-line fingerprint verification. In *International Conference on Pattern Recognition(1996)*, pages 596–600, 1996.
- [34] X. Luo, J. Tian, and Y. Wu. A minutia matching algorithm in fingerprint verification. In *International conference On Pattern Recognition*, pages 833–836, 2000.
- [35] S. Prabhakar and A. K. Jain. Decision-level fusion in fingerprint verification. *Pattern Recognition*, 35(4):861–874, 2002.
- [36] P. Verlinde, G. Chollet, and M. Acheroy. Multi-modal identity verification using expert fusion. *Journal of Information Fusion*, 1(1):17–33, July 2000.
- [37] W.-Y. Yau, K.-A. Toh, X. Jiang, T.-P. Chen, and J. Lu. On fingerprint template synthesis. In *Proceedings of Sixth International Conference on Control, Automation, Robotics and Vision (ICARCV 2000)*, Dec 2000.
- [38] A. K. Jain and A. Ross. Fingerprint mosaicking. In *Acoustics, Speech, and Signal Processing, 2002. Proceedings. (ICASSP '02). IEEE International Conference on*, volume 4, pages IV–4064 – IV–4067, May 2002.
- [39] T. Kamei and M. Mizoguchi. Image filter design for fingerprint enhancement. In *Int. Symp. on Computer Vision*, pages 109–114, 1995.
- [40] C.F. Shu and R.C. Jain. Direct estimation and error analysis for oriented patterns. *Computer Vision Graphics and Image Processing*, 58(3):383–398, November 1993.

- [41] M. W. Hirsch and S. Smale. *Differential equations, dynamical systems, and linear algebra*. Academic Press, 1974.
- [42] J. Weijer, L. J. Vliet, P. W. Verbeek, and M. Ginkel. Curvature estimation in oriented patterns using curvilinear models applied to gradient vector fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9):1035–1042, 2001.
- [43] M.M.S. Chong, R.K.L. Gay, H.N. Tan, and J. Liu. Automatic representation of fingerprints for data compression by b-spline functions. *Pattern Recognition*, 25(10):1199–1210, October 1992.
- [44] A. R. Rao and Ramesh C. Jain. Computerized flow field analysis: Oriented texture fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(7):693–709, 1992.
- [45] C.F. Shu and R.C. Jain. Vector field analysis for oriented patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):946–950, September 1994.
- [46] R.M. Ford, R.N. Strickland, and B. A. Thomas. Image models for 2-d flow visualization and compression. *CVGIP: Graphical Models Image Process.*, 56(1):75–93, 1994.
- [47] R.M. Ford and R.N. Strickland. Nonlinear phase portrait models for oriented textures. In *IEEE Computer Vision and Pattern Recognition*, pages 644–645, 1993.
- [48] R.M. Ford and R.N. Strickland. Representing and visualizing fluid-flow images and velocimetry data by nonlinear dynamical-systems. *CVGIP: Graphical Models and Image Processing*, 57(6):462–482, November 1995.

- [49] M. Kawagoe and A. Tojo. Fingerprint pattern classification. *Pattern Recognition*, 17(3):295–303, 1984.
- [50] A.M. Bazen and S.H. Gerez. Systematic methods for the computation of the directional fields and singular points of fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):905–919, July 2002.
- [51] K. Karu and A. K. Jain. Fingerprint classification. *Pattern Recognition*, 29(3):389–404, March 1996.
- [52] P. Rämö, M. Tico, V. Onnia, and J. Saarinen. Optimized singular point detection algorithm for fingerprint images. In *International Conference on Image Processing 2001*, pages 242–245, 2001.
- [53] K. Rerkrai and V. Areekul. A new reference point for fingerprint recognition. In *International Conference on Image Processing*, pages 499–502, 2000.
- [54] K. Nilsson and J. Bigun. Localization of corresponding points in fingerprints by complex filtering. *Pattern Recognition Letters*, 24:2135–2144, 2003.
- [55] C.-F. Shu, R. Jain, and F. Quek. A linear algorithm for computing the phase portrait of oriented texture. In *International Conference on Computer Vision and Pattern Recognition*, pages 352–357, 1991.
- [56] N.K Ratha, S.Y. Chen, and A.K. Jain. Adaptive flow orientation based feature extraction in fingerprint images. *Pattern Recognition*, 28(11):1657–1672, 1995.
- [57] G. Drets and H. Liljenström. Fingerprint sub-classification and singular point detection. *International Journal of Pattern Recognition and Artificial Intelligence*, 12(4):407–422, 1998.

-
- [58] Federal Bureau of Investigation. *The Science of Fingerprints: Classification and Uses*. U.S. Department of Justice, 1984.
 - [59] J. Zhou and J. Gu. Modeling orientation fields of fingerprints with rational complex functions. *Pattern Recognition*, 37(2):389–391, February 2004.
 - [60] J. Zhou, J. Gu, and D. Zhang. A combination model for orientation field of fingerprints. *Pattern Recognition*, 37(3):543–553, March 2004.
 - [61] J. Li and W. Y. Yau. Prediction of fingerprint orientation. In *International Conference on Pattern Recognition (ICPR2004)*, volume 4, pages 436–439, Aug. 23-26, Cambridge, UK 2004.
 - [62] J. Zhou and J. Gu. A model-based method for the computation of fingerprints' orientation field. *IEEE Transactions on Image Processing*, 13(6):821–835, June 2004.
 - [63] V. I. Arnold. *An Introduction to Analysis(2nd ed.)*. MIT Press, 1973.
 - [64] L. Eldèn. Perturbation theory for the least squares problem with linear equality constraints. *SIAM Journal of Numerical Analysis*, 17:338–350, 1980.
 - [65] A. A. Moenssens. *Fingerprint Techniques*. Chilton, London, 1971.
 - [66] F. Galton. *Finger Prints*. McMillan, 1892.
 - [67] E. Henry. *Classification and Uses of Finger Prints*. Routledge, 1900.
 - [68] G.T. Candela, P.J. Grother, C.I. Watson, R.A. Wilkinson, and C.L. Wilson. Pcasys:a pattern-level classification automation system for fingerprints. In *National Institute of Standards and Technology Technical Report*, 1995.

-
- [69] U. Halici and G. Ongun. Fingerprint classification through self-organizing features maps modified to treat uncertainties. *Processings of the IEEE*, 84(10):1497–1512, October 1996.
- [70] S. Shah and P.S. Sastry. Fingerprint classification using a feedback-based line detector. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, 34(1):85–94, 2004.
- [71] G.L. Marcialis, F. Roli, and P. Frasconi. Fingerprint classification by combination of flat and structural approaches. In *International Conference on Audio- and Video-Based Biometric Person Authentication*, pages 241–246, 2001.
- [72] Y. Yao, G.L. 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 Recognition*, 36(2):397–406, February 2003.
- [73] M. Ballan, F. A. Sakarya, and B. L. Evans. A fingerprint classification technique using directional images. In *Proceedings of IEEE Asilomar Conference on Signals, Systems, and Computers*, volume 1, pages 101–104, 1997.
- [74] L. Hong and A.K. Jain. Classification of fingerprint images. In *Proceedings of 11th Scandinavian Conference on Image Analysis (SCIA99)*, 1999.
- [75] B.-H. Cho, J.-S. Kim, J.-H. Bae, I.-G. Bae, and K.-Y. Yoo. Fingerprint image classification by core analysis. In *Signal Processing Proceedings, 2000. WCCC-ICSP 2000. 5th International Conference on*, volume 3, pages 1534–1537, 2000.
- [76] C. Klimanee and D. T. Nguyen. Classification of fingerprints using singular points and their principal axes. In *International Conference on Image Processing*, pages 849–852, 2004.

- [77] S.M. Mohamed and H. Nyongesa. Automatic fingerprint classification system using fuzzy neural techniques. In *Fuzzy Systems, 2002. FUZZ-IEEE'02. Proceedings of the 2002 IEEE International Conference on*, volume 1, pages 358 – 362, 2002.
- [78] C.L. Wilson, G.T. Candela, and C.I. Watson. Neural network fingerprint classification. *Journal of Artificial Neural Networks*, 1(2):203–228, 1994.
- [79] M.M.S. Chong, T.H. Ngee, L. Jun, and R.K.L. Gay. Geometric framework for fingerprint image classification. *Pattern Recognition*, 30(9):1475–1488, September 1997.
- [80] A. Senior. A combination fingerprint classifier. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):402–421, 1999.
- [81] A.K. Jain and S. Minut. Hierarchical kernel fitting for fingerprint classification and alignment. In *International Conference on Pattern Recognition(ICPR02)*, pages II: 469–473, 2002.
- [82] J.H. Chang and K.C. Fan. A new model for fingerprint classification by ridge distribution sequences. *Pattern Recognition*, 35(6):1209–1223, June 2002.
- [83] S. C. Dass and A. K. Jain. Fingerprint classification using orientation field flow curves. In *the 4th Indian Conference on Computer Vision, Graphics and Image Processing*, 2004.
- [84] Y. Xu, S. X. Yang, G. He, and X. Zhang. Macroscopic curvature-based fingerprint feature extraction and analysis. In *DCDIS 4th International Conference on Engineering Applications and Computational Algorithms*, 2005.

-
- [85] M. Neuhaus and H. Bunke. A graph matching based approach to fingerprint classification using directional variance. In *the 5th Audio- and Video-Based Biometric Person Authentication*, pages 191–200, 2005.
 - [86] B. Moayer and K. S. Fu. A syntactic approach to fingerprint pattern recognition. *Pattern Recognition*, 7(1-2):1–23, June 1975.
 - [87] B. Moayer and K.S. Fu. An application of stochastic languages to fingerprint pattern recognition. *Pattern Recognition*, 8(3):173–179, July 1976.
 - [88] T. Kamei and M. Mizoguchi. Fingerprint preselection using eigenfeatures. In *IEEE Computer Vision and Pattern Recognition (CVPR98)*, pages 918–923, 1998.
 - [89] M. Kamijo. Classifying fingerprint images using neural network: Deriving the classification state. In *Proceedings of the Third International Conference On Neural Network*, pages 1932–1937, 1993.
 - [90] R. Cappelli, D. Maio, and D. Maltoni. A multi-classifier approach to fingerprint classification. *Pattern Analysis & Applications*, 5(2):136–144, 2002.
 - [91] R. Cappelli, D. Maio, and D. Maltoni. Fingerprint classification based on multi-space kl. In *Workshop on Automatic Identification Advances Technologies (AutoID'99)*, pages 117–120, October 1999.
 - [92] A.P. Fitz and R.J. Green. Fingerprint classification using a hexagonal fast fourier-transform. *Pattern Recognition*, 29(10):1587–1597, October 1996.
 - [93] C.H. Park and H. Park. Fingerprint classification using fast fourier transform and nonlinear discriminant analysis. *Pattern Recognition*, 38(4):495–503, 2005.

- [94] F. Su, J.-A. Sun, and A. Cai. Fingerprint classification based on fractal analysis. In *Signal Processing Proceedings, 2000. WCCC-ICSP 2000. 5th International Conference on*, volume 3, pages 1471–1474, 2000.
- [95] J. Li, W.Y. Yau, and H. Wang. Constrained nonlinear models of fingerprint orientations with prediction. *Pattern Recognition*, 39(1):102–114, January 2006.
- [96] J. Bigun, T. Bigun, and K. Nilsson. Recognition by symmetry derivatives and the generalized structure tensor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:1590–1605, 2004.
- [97] K. Nilsson and J. Bigun. Prominent symmetry points as landmarks in fingerprint images for alignment. In *International Conference on Pattern Recognition(ICPR2002)*, volume III, pages 395–398. IEEE computer society, Aug. 11-15, Quebec, Canada 2002.
- [98] S. R. Gunn. *Support vector machines for classification and regression*. Univerity of Sounthampton, 1998.
- [99] L. I. Kuncheva. A theoretical study on six classifier fusion strategies. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):281–286, 2002.
- [100] J. Ma, Y. Zhao, and S. Ahalt. *OSU SVM Classier Matlab Toolbox (ver 3.00)*. http://eewww.eng.ohiostate.edu/~maj/osu_svm/, 2002.
- [101] K.A. Toh, W. Y. Yau, and X.D. Jiang. A reduced multivariate polynomials model for multi-modal biometrics and classifiers fusion. *IEEE Transactions on Circuits and Systems for Video Technology (Special Issue on Image- and Video-Based Biometrics)*, 14(2):224–233, Feb. 2004.

-
- [102] M.S. Pattichis, G. Panayi, A.C. Bovik, and S.P. Hsu. Fingerprint classification using an am-fm model. *IEEE Transactions on Image Processing*, 10(6):951–954, June 2001.
 - [103] A. Lumini, D. Maio, and D. Maltoni. Continuous versus exclusive classification for fingerprint retrieval. *Pattern Recognition Letter*, 18(10):1027–1034, 1997.
 - [104] B. Bhanu and X. Tan. Fingerprint indexing based on novel features of minutiae triplets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):616–622, 2003.
 - [105] S. T. Acton, D. P. Mukherjee, J. P. Havlicek, and A. C. Bovik. Oriented texture completion by am-fm reaction-diffusion. *IEEE Transaction on Image Processing*, 10(6):885–896, Jun 2001.