The Hong Kong Polytechnic University

Department of Computing

Personal Authentication Using Finger-Knuckle-Print

by

ZHANG Lin

A thesis submitted in partial fulfillment of the requirements

for the Degree of Doctor of Philosophy

January 2011

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree of diploma, except where due acknowledgement has been made in the text.

__ _ (Signed)

ZHANG Lin (Name of Student)

Abstract

Biometrics, which is the discipline of recognizing a person's identity based on his/her physical or behavioural characteristics, has attracted much attention in the recent decade due to its numerous applications. Each traditional biometric identifier has its own advantages and disadvantages. Thus, researchers have never stopped searching for new kinds of biometric identifiers. By observing that the texture pattern produced by bending the finger knuckle is highly distinctive, in this work we present a new biometric authentication system using finger-knuckle-print (FKP) imaging. Various aspects of this system are thoroughly investigated and discussed in this thesis.

Finger-knuckle-print, which refers to the inherent skin patterns of the outer surface around the phalangeal joint of one's finger, is a new member of the biometrics family. It has high capability to discriminate different individuals. As a kind of hand-based biometrics, FKP has the merit of high user friendliness. Compared with the other traditional hand-based biometric identifiers, for example the fingerprint, FKP may have some special advantages. At first, it is not easy to be abraded and forged since people usually hold stuffs with the inner side of the hand. Moreover, unlike the fingerprint, there is no stigma of criminal investigation associated with the FKP, so it can have a high user acceptance. These characteristics make the FKP have a great potential to be a widely accepted promising biometric identifier.

Our FKP recognition system comprises four major components: FKP image acquisition, ROI (region of interest) extraction, feature extraction, and feature matching. At first, a specially designed FKP image acquisition device is established. It is composed of a finger bracket, a ring LED light source, a lens, and a CCD camera. Such a device can be conveniently used and can capture high quality FKP images. With the developed acquisition device, a large FKP database containing 7920 samples collected from 660 fingers is established and now it is publicly available.

After an FKP image is captured, a region of interest (ROI) needs to be cropped from the original image for the further feature extraction and matching. Such a pre-processing step can reduce the data amount in the feature extraction and matching stage and can abate the influence of variations of the FKPs. To this end, an efficient FKP ROI extraction algorithm is proposed based on the intrinsic characteristics of FKP images.

As in any pattern classification task, the feature extraction and matching plays a key role in our FKP-based personal authentication system. To this end, we have developed and examined a couple of different methods. At first, the performances of several state-of-the-art coding based methods are evaluated, including CompCode, RLOC, and OrdinalCode. After that, a novel coding-based feature extraction and matching method, namely ImCompCode&MagCode is proposed, which is an extended version of CompCode. Moreover, we propose two more efficient and compact coding based feature extraction and matching approaches, RCode1 and RCode2, using Riesz transforms.

In fact, coding based feature extraction and matching methods make use of local information of images. Actually, global information hidden in images can also be exploited for recognition. Based on this belief, we have developed another FKP matching method by matching the Fourier transform coefficients of two FKP images using the phase-only correlation technique.

Furthermore, based on the results of psychophysics and neurophysiology studies that both local and global information is crucial for the image perception, we present an effective FKP recognition scheme by extracting and assembling local and global features of FKP images. Specifically, we use the local orientation, the local phase, and the local phase congruency as the local features and they can be extracted by using Gabor filters. By increasing the scale of Gabor filters to infinity, actually we get the Fourier transform of the image, and hence the Fourier transform coefficients of the image can be taken as the global features. Such kinds of local and global features are naturally linked via the framework of the time-frequency analysis.

All the developed recognition methods are thoroughly investigated on our established benchmark FKP database. And it needs to be noted that the feature extraction and matching methods developed in this thesis can also be applied to some other biometrics systems, e.g., the palmprint recognition system.

Another contribution of this thesis is that the developed technologies have been implemented in a standalone embedded FKP recognition system, which can be readily used in practice. Such a system is the first of its kind.

Publications

The following papers, published or in press, are the partial output of my PhD studies in PolyU.

1. Lin Zhang, Lei Zhang, Hailong Zhu, and David Zhang, Ensemble of local and global information for finger-knuckle-print recognition, *Pattern Recognition*, vol. 44, no. 9, pp. 1990-1998, 2011.
2. Lin Zhang, Lei Zhang, Hailong Zhu, and David Zhang, Online finger-knuckle-print verification for personal authentication, *Pattern Recognition*, vol. 43, no. 7, pp. 2560-2571, 2010.
3. Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang, FSIM: a feature similarity index for image quality assessment, *IEEE Transactions on Image Processing*, accepted.
4. Bob Zhang, Lin Zhang, Lei Zhang, and Fakhri Karray, Retinal vessel extraction by matched filter with first-order derivative of Gaussian, *Computers in Biology and Medicine*, vol. 40, no. 4, pp. 438-445, 2010.
5. Lin Zhang, Lei Zhang, Zhenhua Guo, and David Zhang, Monogenic-LBP: a new approach for rotation invariant texture classification, *Proceedings of the International Conference on Image Processing*, pp. 2677-2680, 2010.
6. Lin Zhang, Lei Zhang, and Xuanqin Mou, RFSIM: a feature based image quality assessment metric using Riesz transforms, *Proceedings of the International Conference on Image Processing*, pp. 321-324, 2010.
7. Meng Yang, Lei Zhang, Lin Zhang, and David Zhang, Monogenic binary pattern (MBP): a novel feature extraction and representation model for face recognition, *Proceedings of the International Conference on Pattern Recognition*, pp. 2680-2683, 2010.
8. Lin Zhang, Lei Zhang, and David Zhang, MonogenicCode: a novel fast feature coding

algorithm with applications to finger-knuckle-print recognition, *Proceedings of the International Workshop on Emerging Techniques and Challenges for Hand-Based Biometrics*, 2010.

9. Lin Zhang, Lei Zhang, and David Zhang, Finger-knuckle-print: a new biometric identifier, *Proceedings of the International Conference on Image Processing*, pp. 1981-1984, 2009.

10. Lin Zhang, Lei Zhang, and David Zhang, Finger-knuckle-print verification based on band-limited phase-only correlation, *Proceedings of the International Conference on Computer Analysis of Images and Patterns*, pp. 141-148, 2009.

11. Lin Zhang, Lei Zhang, and David Zhang, A multi-scale bilateral structure tensor based corner detector, *Proceedings of the Ninth Asian Conference on Computer Vision*, pp. 618-627, 2009.

Acknowledgements

The three-year doctoral training has gone so rapidly and is now meeting its end. I happily arrive at the moment to summarize my work and life during the past three years. I would like to take this opportunity to express my sincere thanks to everyone who generously gave me their care and support during my PhD study.

First of all, my deep gratitude goes to my supervisor, Dr. Lei Zhang for his rigorous supervision of my research. His scientific vision, passion, high criterion, and detailed comments have been a major source of inspiration and motivation for my work. Time after time, he showed me how to express ideas clearly and write academic papers deeply. His extraordinary enthusiasm for research impacts me greatly. What I have learned and experienced during the time I spent in his laboratory will benefit me much in the future.

Furthermore, I acknowledge the support and help from Prof. David Zhang, Dr. Hailong Zhu, Prof. Yongping Zheng, Prof. Jane You, and Dr. Guangming Lu.

I would like to thank my classmates, colleagues, and visiting scholars in the BRC lab, Dr. Qijun Zhao, Dr. Zhenhua Guo, Nan Luo, Xiaofeng Qu, Dr. Wei Li, Xingzheng Wang, Yahui Liu, Bo Shen, Prof. Xiangchu Feng, Dr. Jifeng Ning, Dr. Jian Bai, Dr. Guojun Liu, Dr. Yafeng Li, Xuande Zhang, Dr. Shigang Liu, Dr. Wei Jia, Dr. Qinghua Hu, Dr. Qin Li, Dongmin Guo, Jin Xie, Feng Liu, Zhizhao Feng, Kaihua Zhang, Dr. Min Zhang, Po Peng, Dr. Weisheng Dong, Meng Yang and all other members of the PolyU BRC group that I cannot enumerate here. I thank them for interesting discussions and suggestions on my work. In addition, we share the pain and pleasure of the PhD studies or working at The Hong Kong Polytechnic University.

My deepest gratitude goes to my family for their mental support and encouragement, my parents Shujing Zhang and Guiqin Ren, my parents-in-law Chuanmin Shen and Yuzhu Wang, and my beautiful and considerate wife Ying Shen. They will not read this thesis, but without their love and support I can’t complete this journey.

Table of Content

List of Figures

List of Tables

Chapter 1. Introduction

1.1 A Brief Introduction of Biometrics

Recognizing the identity of a person with high confidence is a critical issue in various applications, such as e-banking, access control, passenger clearance, etc. The need for reliable user authentication techniques has significantly increased in the wake of heightened concerns about security, and rapid advancement in networking, communication and mobility [1].

The objective of personal authentication is to determine or confirm the identity of individuals such that the right person is found out from a number of suspects, and the requested services or facilities are accessed by a legitimate user, etc. Traditionally, personal authentication is fulfilled based on what the person has (e.g. keys and ID cards) or what the person knows (e.g. passwords). These approaches however have at least the following two drawbacks. 1) Both the tokens the user has and the knowledge the user knows can be lost, forgotten or stolen. 2) They have the problem of repudiation, e.g., a person accesses a certain resource and later claims that another person must have used it under counterfeited credentials.

Biometrics based methods, which use unique inherent physical or behavioural characteristics of human beings, can solve above problems [1-5]. It is well known that human beings instinctively make use of some body characteristics, e.g. face, gait, or voice, to recognize each other. Because the biometric traits are inherent in people, one is not bothered by forgetting or losing them or having them stolen and consequently he/she can not deny his/her ever use of his/her biometric traits. Biometrics is the most accurate form of identifiers and, if used properly, can greatly simplify life [3].

A biometric system is essentially a pattern recognition system that acquires biometric data from an individual, extracts a salient feature set from the data, compares this feature set against the feature set(s) stored in the database, and takes an action based on the result of the comparison. Figure 1.1 shows the generic architecture of a typical biometric system [6].

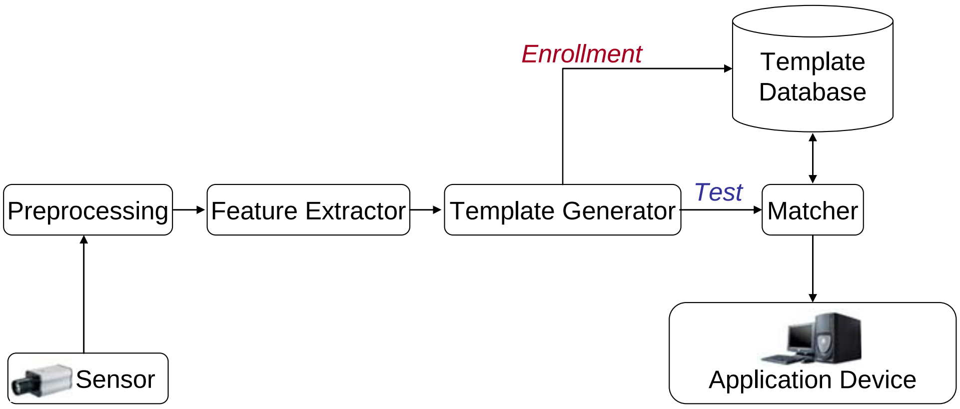


Figure 1.1 A typical framework of a biometric system. (The figure is regenerated from [6].)

Table 1.1 Total global biometric market revenue by technology, 2005-2012. (Adapted from BCC Research: http://www.bccresearch.com/report/IFT042B.html)

	2005 ($ million)	2006 ($ million)	2007 ($ million)	2012 ($ million)
Fingerprint	739.5	955.5	1,269.0	2,698.0
Face	236.4	327.6	459.0	1,334.8
Hand Geometry	118.9	165.8	243.0	752.6
Others	355.2	501.1	729.0	2314.6
Total	1,450.0	1,950.0	2,700.0	7,100.0

Biometrics has already being used in many different applications, such as electronic data security, internet access, computer network login, ATM or credit card use, physical assess control, e-commerce, mobile phone, PDA, social security, passport control, corpse identification, criminal investigation, parenthood determination, national ID card, driver's license, etc [1]. In Hong Kong, the government has been using the fingerprint recognition system as the automated passenger clearance system (e-channel) since 2004 [7]. According to the report of BCC Research [8] as shown in Table 1.1, the global market for biometrics increased from $1450 million in 2005 to an estimated $7100 million by the end of 2012.

1.2 Glossary

In order to make the reading of this thesis smoothly, some glossaries used throughout this thesis are explained here. These definitions are extracted from [1-5, 9].

Enrolment

With the biometric system, the administrator can collect biometric samples from a person. Then, the biometric features can be extracted from the collected samples and then be stored in the system's database. These features are used as reference templates representing that person's identity. Such a process is defined as "enrolment".

Verification and Identification

According to the application context, personal authentication has two different modes: verification and identification. Under the verification mode, the biometric system validates a user's identity by comparing his/her captured biometric data with his/her own biometric template(s) previously enrolled and stored in the database. Usually, in such a system, the user claims an identity, and then the system conducts a one-to-one matching to determine whether such a claim is true or not. In the identification mode, the system recognizes a user's identity by matching his/her supplied biometric data with the templates of all the users registered in the database. That means the system conducts a one-to-many comparison to recognize who the person is.

False Accept Rate (FAR), False Reject Rate (FRR), and Equal Error Rate (EER)

The degree of similarity between two biometric feature sets is represented by a similarity score. A matching score is regarded as a genuine score if it is a result of mathing two samples of the same biometric trait of the same user. Otherwise, the matching score is considered as an impostor score. The False Accept Rate (FAR) of a biometric system is defined as the fraction of impostor scores that exceed a preset threshold η. In a similar way, the False Reject Rate (FRR) of a biometric system is defined as the fraction of genuine scores that fall below the threshold η. By modulating the value of the threshold η, the FRR and the FAR values can be changed. When the threshold η is set so that the FRR is approximately equal to the FAR, the error rate (FRR or FAR) occurring is known as Equal Error Rate (EER). A low EER value, therefore, indicates better recognition performance.

Receiver Operating Characteristic (ROC) Curve

By adjusting the value of the threshold η, a curve can be created, which is a plot of FRR against FAR for all possible thresholds. Such a curve is defined as Receiver Operating Characteristic (ROC) curve. A typical ROC curve of a biometric system is shown in Figure 1.2.

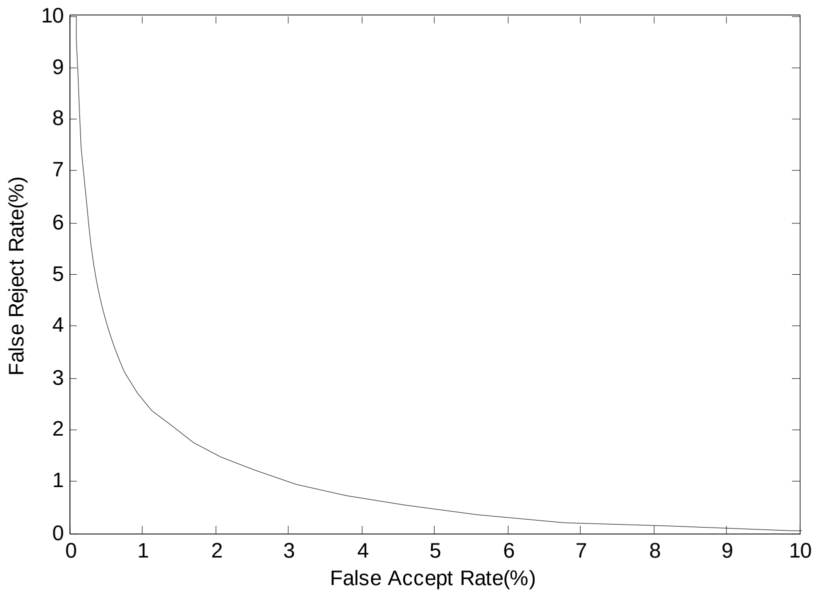


Figure 1.2 An example of an ROC curve.

Decidability Index *d'*

d' is used to measure how well the genuine and the imposter distributions are separated. *d'* is defined as

$$d' = \frac{|\mu_1 - \mu_2|}{\sqrt{(\sigma_1^2 + \sigma_2^2)/2}} \tag{1-1}$$

where μ_1 (μ_2) is the mean of the genuine (imposter) matching scores and σ_1 (σ_2) is the standard deviation of the genuine (imposter) matching scores.

1.3 Overview of the Biometric Identifiers

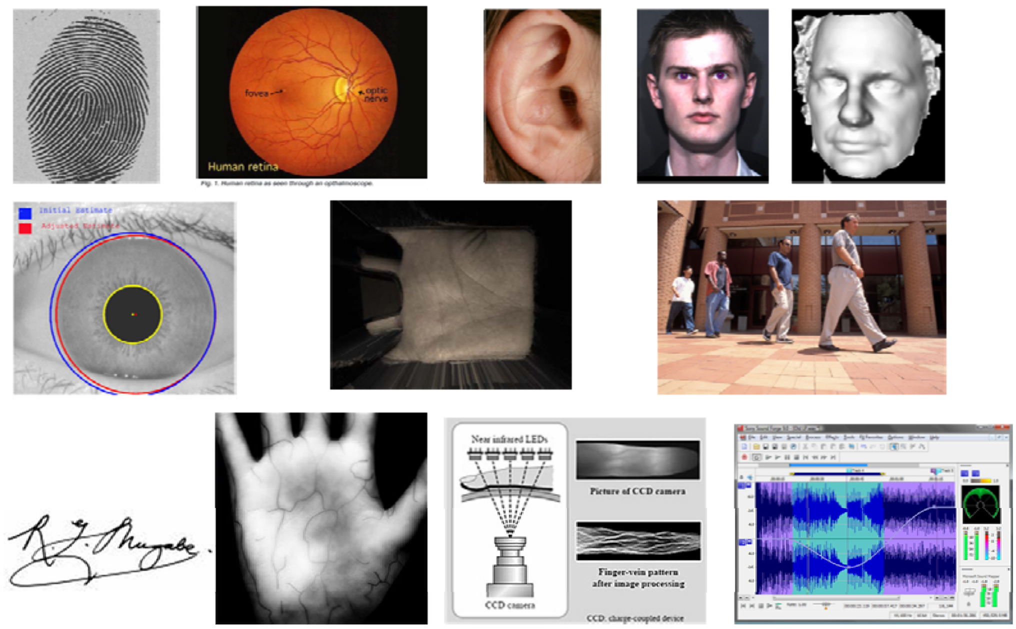


Figure 1.3 Kinds of biometric identifiers.

Currently, there are a number of different biometric identifiers being used in various applications in practice. In past decades, researchers have exhaustively investigated many biometric identifiers, including face [10-22], iris [23-31], retina [32, 33], ear [34, 35], tongueprint [36, 37], fingerprint [38-46], palmprint [47-66], hand geometry [67-70], hand vein [71, 72], finger vein [73-75], finger surface [76-82], inner knuckle print [83, 84], footprint [85-88], voice [89, 90], gait [91], and signature [92, 93], etc. Images of several biometric identifiers are shown in Figure 1.3.

However, each biometric has its own pros and cons. As identified by Jain *et al.*, there are seven factors [9] that can determine the suitability of a physical or behavioural trait to be used as a biometric identifier, including:

- Universality. It is desired that every individual assessing the application should have the biometric trait that is going to be used in the system. Universality describes how commonly a biometric identifier can be found in an individual and how readily it can be used.
- Uniqueness. It is the degree to which a biometric trait differentiates one individual from another.
- Permanence. The biometric trait should be sufficiently invariant with respect to the matching scheme over a period of time.
- Measurability. It indicates the practicality and ease of acquiring a biometric trait using suitable devices.
- Performance. It indicates the biometric system's accuracy, robustness, and speed.
- Acceptability. It indicates the degree of the target population to accept the biometric trait going to be used.
- Circumvention. This refers to how hard it is to spoof a biometric system.

According to these measures, no biometric identifier can be expected as ideal. No one has yet devised a perfect biometric system. There are a variety of biometric techniques, each with its own advantages and disadvantages. The following are just some representative examples [1, 5, 75].

Fingerprint is the oldest and the most widely used biometric modality today. Fingerprint recognition has merits of low-cost and high accuracy. Figure 1.4 illustrates two applications of the fingerprint. However, there are some limitations of the fingerprint scanning that cannot be ignored. Due to various causes, such as genetic factors, aging, environmental or occupational reasons, 2~5% of the population cannot provide high quality fingerprint images for the automated authentication [94]. Fingerprint ridges deteriorate with age. In addition, some working activities, such as washing hands many times a day or constantly placing fingers in corrosive chemicals, can erode the person's fingerprints. Another disadvantage of fingerprint is that it has a latency property. That is we will leave fingerprint remains on the surface we touch. This can lead to loss of the personal privacy and can decrease the security of the fingerprint-based biometric systems. That is why some people express a strong aversion when collecting their fingerprint samples.

(a) (b)

Figure 1.4 Applications of fingerprint recognition systems. (a) The system is installed on a door to act as an access control system; (b) the system is installed on a car to identify the identity of the car's owner.

Figure 1.5 Audiences were undergoing verification checks at face recognition checkpoints before entering the National Stadium on August 8, 2008 [95].

Face recognition is a non-intrusive approach and can be considered as the most convenient and user friendly modality. Automated face recognition technology has been commercialized since mid-1990s and in the past decade the technology has made rapid advancements. It has been widely used in various occasions. For example, in Olympic Games 2008 held in Beijing, China, a face recognition system developed by the Institute of Automation Chinese Academy of Sciences was used to aid the access control. Figure 1.5 shows that audiences were undergoing verification checks at face recognition checkpoints before entering the National Stadium on August 8, 2008 [95]. It is the first time that such technology was adopted as security measures in the Olympic history. However, the existing face based systems still have difficulty in matching face images collected from different views, under different lighting conditions, having different expressions, or at different capturing sessions. Moreover, some people worry about the privacy disclosure raised by using face recognition systems since sometimes the face recognition can occur even when the subject does not notice it.

Figure 1.6 The outlook of a hand geometry recognition system.

Hand geometry based biometric systems, which focus on measuring the physical structure (e.g. size of the palm, lengths and widths of the fingers) of an outstretched hand, are among the most commonly used systems. Figure 1.6 shows the outlook of a hand geometry recognition system. Hand geometry recognition is relatively simple, easy to use, and inexpensive. However, its accuracy is not satisfactory and thus it is not appropriate for most identification applications. Furthermore, hand geometry features vary over time, more than most other biometric identifiers.

(a) (b)

Figure 1.7 Two types of iris recognition systems.

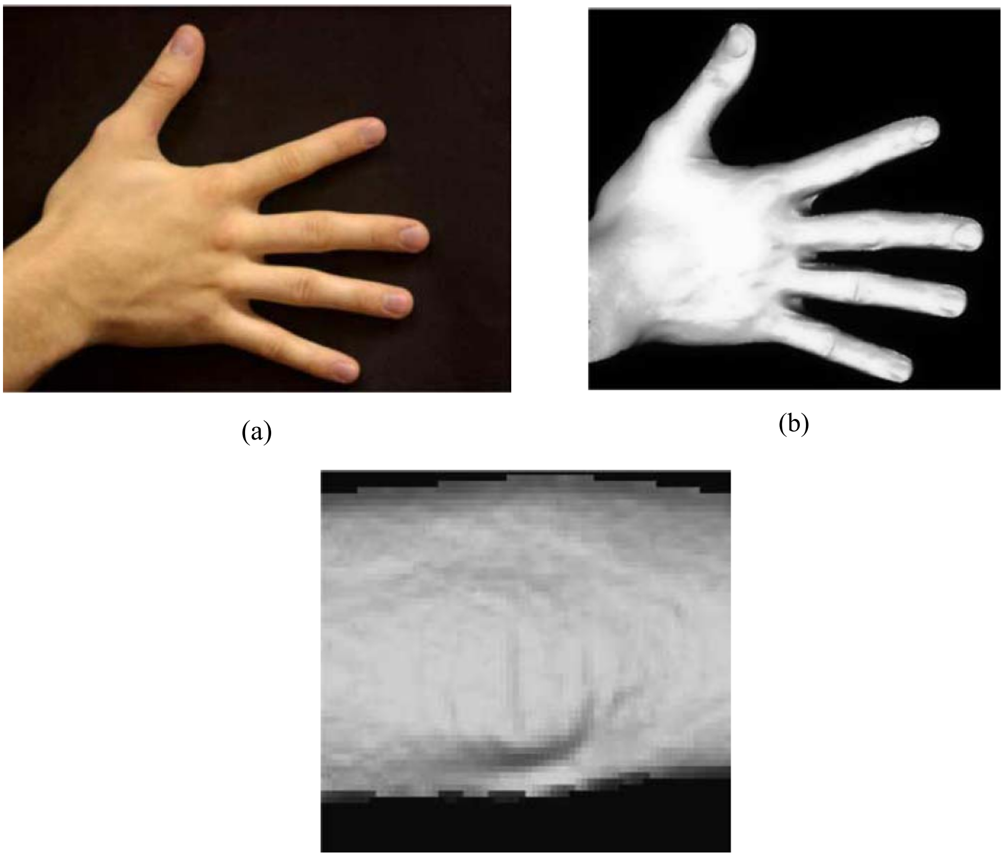
Iris is a popular and reliable biometric trait. Figure 1.7 shows two types of iris recognition systems. Iris features are highly stable over time and they are less subject to abrasion or injury than most other body traits. However, the associated devices are expensive, and some people are reluctant to accept it because they worry about possible damage to their eyes. In addition, the collection of an iris image requires more training and attentiveness than most other biometric modalities.

Voice verification is a very natural way by which human beings recognize each other. Now voice has also been exploited as a biometric identifier for the use of personal authentication. Voice is a combination of physical and behavioural biometric traits [89]. Voice recognition has several advantages: 1) it has high user acceptance; 2) the sample acquisition device used in voice recognition is contact less; and 3) the sensors, such as telephones and PC microphones, are commonly available. However, voice recognition is not sufficiently distinctive, especially on large databases. It is not uncommon that some people may have similar voices. Moreover, voices of people will change over time and depend on many environmental factors, such as health, stress, the emotional state, etc.

1.4 Finger-knuckle-surface as a Biometric Identifier

Researchers have never stopped investigating new kinds of biometric identifiers. Among various kinds of biometric identifiers, hand-based ones, such as palmprint, fingerprint, and hand geometry, have been attracting considerable attention over recent years. The popularity of hand-based biometrics should be attributed to its high user acceptance. Recently, it has been found that the image pattern of skin folds and creases in the outer finger knuckle surface is highly unique and thus can serve as a distinctive biometric identifier. Compared with fingerprint, the finger knuckle surface has some advantages as a biometric identifier. At first, it is not easy to be abraded since people usually hold stuffs with the inner side of the hand. In addition, unlike the use of fingerprint, there is no stigma of criminal investigation associated with the finger knuckle surface, so it can have a high user acceptance [80]. Furthermore, it is not easy to be forged since people rarely leave remains of finger surfaces on stuffs. Thus, the finger knuckle feature has a great potential to be widely accepted as a biometric identifier. Some researchers have already done salient works in this field.

Woodard and Flynn [76-78] are among the first scholars who exploited the use of finger knuckle surface in biometric systems. They set up a 3D finger back surface database with the Minolta 900/910 sensor. The sensor captured both a 640×480 range image and a registered 640×480 24-bit colour intensity image. Before the data collection, a black cloth was used to cover a flat wall. Black cloth was used as the background to simplify the hand data segmentation task. Before the samples were collected, the subject was instructed to put his/her hand flatly against the wall with the fingers naturally spread. Figure 1.8 shows some sample images collected by their system. Figure 1.8(a) is a sample colour image of a hand. Figure 1.8(b) is the corresponding range image of the same hand. Figure 1.8(c) depicts the surface detail around the finger knuckle area. For feature extraction, they used the curvature based shape index to represent the finger back surface. Woodard's work makes a good effort to validate the uniqueness of outer finger surface as a biometric characteristic. However, they did not provide a practical solution in establishing an efficient system using the outer finger surface features. The cost, size and weight of the Minolta 900/910 sensor limit the use of it in a practical biometric system, and the time-consuming 3D data acquisition and processing limit its use in real-time applications. In addition, they did not fully exploit the finger knuckle texture information in feature extraction.



(a) (b)

(c)

Figure 1.8 Sample images collected by Woodard *et al*. [77, 78]. (a) is a colorful intensity image; (b) is the corresponding range image; (c) shows details of a range image.

(a)

(b)

Figure 1.9 (a) The image acquisition device used in Kumar *et al.*'s system [80]; (b) a sample image collected by the device shown in (a).

Later, Kumar and Ravikanth [79, 80] proposed another approach to personal authentication by using 2-D finger-back surface imaging. They developed a system to capture hand-back images and then extracted the finger knuckle areas by some pre-processing steps. Figure 1.9(a) shows the outlook of the imaging device used in their system and Figure 1.9(b) shows a sample image captured by their device. The subspace analysis methods such as PCA, LDA and ICA were combined to do feature extraction and matching. With Kumar's design, the acquisition device is doomed to have a large size because nearly the whole hand back area has to be captured, despite the fact that the finger knuckle area only occupies a small portion of the acquired image. Furthermore, subspace analysis methods may be effective for face recognition but they may not be able to effectively extract the distinctive line and junction features from the finger knuckle surface.

1.5 Contributions and the Outline of the Thesis

From the above introduction, we can find that although the finger back surface has a great potential to be a promising biometric identifier, its research is still in infancy. At present, there are only a few tentative works focusing on exploring finger back surface features for the purpose of personal authentication and the results are far from perfection. Particularly, the prototype systems created in [77] and [80] cannot be used in practice. At first, there is no specially designed device for conveniently and effectively collecting images from finger back surfaces. Secondly, the collecting devices used in those prototypes are extremely large which cannot be readily used in practice. Furthermore, the unique characteristics of finger back surface images have not been fully exploited when designing feature extraction and matching algorithms. Thus, there is still large room left to fully explore finger back surface features for the personal authentication.

In this thesis, we refer image patterns of skin folds and creases in the outer finger knuckle surface as finger-knuckle-print, or FKP for short. This thesis presents a practical brand-new FKP-based personal authentication system, which represents the first of its kind. Partial research outputs covered in this thesis have been published in several papers [96-104]. The developed system can work alone or can also be incorporated in a multi-modal biometrics system. Such a system comprises four major components: a device for FKP image acquisition, ROI (region of interest) extraction, feature extraction, and feature matching. To test its performance, a large FKP image database has been established. In the subsequent chapters, different aspects of our FKP recognition system will be described in detail.

Chapter 2 at first presents the overall structure of the proposed FKP recognition system. Then, the specially designed FKP imaging device will be introduced. Unlike the systems in [77] and [80] which first capture the image of the whole hand and then extract the finger or finger knuckle surface areas, the proposed device captures the image around the finger knuckle area of a finger directly, which largely simplifies the following preprocessing steps. Meanwhile, with such a design the size of the imaging system can be greatly reduced, which improves much its applicability. Since the finger knuckle will be slightly bent when being imaged in the proposed system, the inherent finger knuckle print patterns can be clearly captured and hence the unique features of FKP can be better exploited. In order to evaluate the performance of the whole system, a database has been established using the developed FKP image acquisition device. This database contains 7,920 images collected from 660 fingers. The data pre-processing is also described in Chapter 2. An efficient FKP ROI (region of interest) extraction algorithm is proposed based on the intrinsic characteristics of FKP images.

As in any pattern classification task, the feature extraction and matching plays a key role in our FKP-based personal authentication system. To this end, we have developed a couple of different methods and they are covered in Chapters 3, 4, and 5. In Chapter 3, at first, the performances of several state-of-the-art coding based methods are examined, including CompCode [51], RLOC [59], and OrdinalCode [53]. Then a new coding method, namely ImCompCode&MagCode is presented, which is an extended version of CompCode. In ImCompCode&MagCode, not only the orientation information but also the magnitude information is considered. Moreover, two more efficient and compact coding based feature extraction and matching approaches based on Riesz transforms [105] are presented.

In fact, coding based feature extraction and matching methods make use of the local information of images. Actually, the global information hidden in images can also be exploited for recognition. Based on this belief, in Chapter 4, an FKP recognition method by matching the Fourier transform coefficients between two images using the phase-only correlation technique [106] is presented.

In the literature of psychophysics and neurophysiology, many studies have shown that both local and global information is crucial for the image perception and recognition of human beings and they play different but complementary roles [21]. A global feature reflects the holistic characteristics of the image and is suitable for coarse representation, while a local feature encodes more detailed information within a specific local region and is appropriate for finer representation. In Chapter 5, with the belief that the recognition accuracy could be much improved by the fusion of local-global features, we propose a novel effective unified local-global feature extraction and fusion scheme for the verification of FKP images. The features used in our scheme are some low level image features defined under the time-frequency analysis framework. Specifically, we use the local orientation, the local phase, and the local phase congruency as the local features and they reflect different aspects of information within a local patch and none of them can be covered by the others. It will be seen that these three kinds of local features depend on the local time-frequency analysis of the image. In practice, quadrature pair of filters, such as complex Gabor [107] or log-Gabor filters [108], can be used to do the local time-frequency analysis. Suppose that we use the complex Gabor transform to extract the local features. By increasing the scale of the complex Gabor filters, more and more global information will be involved, yet the characterization of image local structures will be rapidly weakened. Particularly, if the scale of the Gabor transform is increased to infinity, the Gabor transform can be reduced to the Fourier transform of the whole image. In this case, no image local information can be extracted but we can get the finest resolution for image global frequency analysis. Thus, the Fourier transform coefficients are naturally taken as global features. With the global Fourier features, the alignment between intra-class FKP ROIs can also be refined. At the matching stage, four matching distances can be computed by comparing the local features and the global features separately. Finally, the four matching distances are fused according to some fusion rule to get the final matching distance. The experimental results are very promising, demonstrating the effectiveness of the proposed scheme. The effectiveness of this local-global feature scheme is also corroborated on a benchmark palmprint database.

In Chapter 6, based on the above studies, a standalone embedded FKP identification system is implemented. It is based on the OMAP3530 processor and WinCE 6.0. Such a system can be readily used in practice. Some issues regarding the hardware and software design are discussed there.

Finally, the summary and the outlook of this thesis are given in Chapter 7. Open problems and future work are also discussed there.

Chapter 2. Overview of the FKP Recognition System

In this chapter, the overall structure of our proposed finger-knuckle-print (FKP) recognition system will be described at first. Then, the specially designed FKP imaging device will be presented in detail. The FKP dataset collected by using the developed acquisition device will also be introduced. After that, the FKP image pre-processing and the ROI extraction algorithm will be given.

2.1 System Structure and the FKP Imaging Device

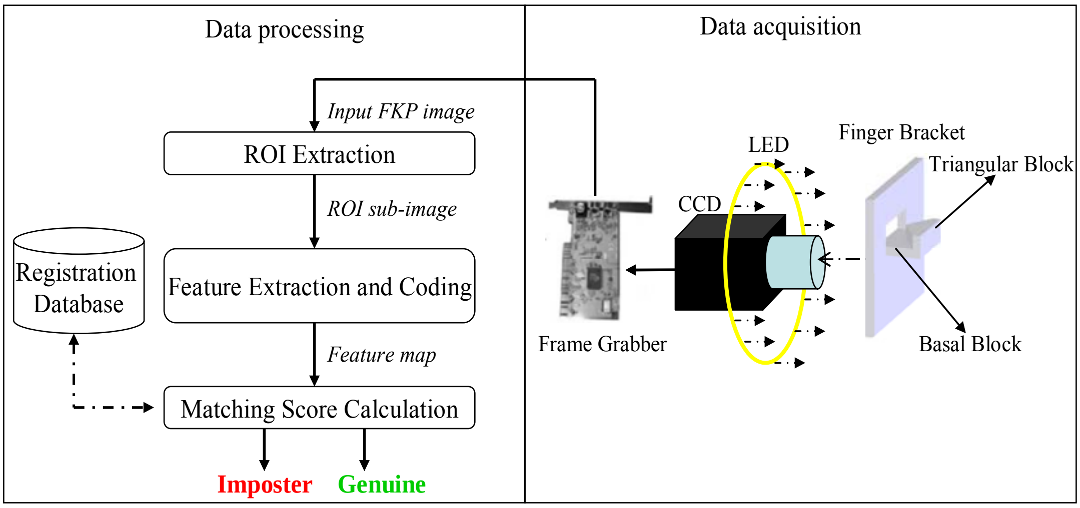
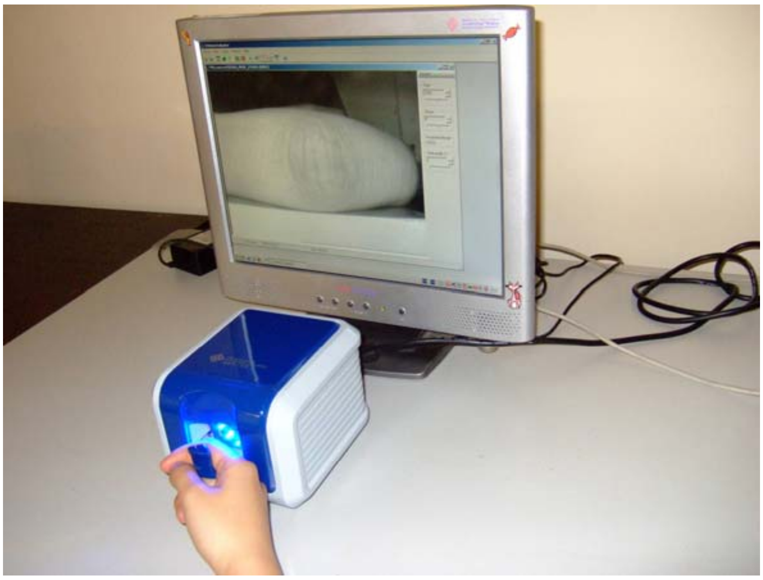


Figure 2.1 Structure of the proposed FKP-based personal authentication system. The whole system is composed of a data acquisition module and a data processing module.

The schematic diagram of our FKP based personal authentication system is shown in Figure 2.1. The system is composed of a data acquisition module and a data processing module. The data acquisition module is composed of a finger bracket, a ring LED light source, a lens, a CCD camera and a frame grabber. The captured FKP image is inputted to the data processing module, which comprises three basic steps: ROI (region of interest) extraction, feature extraction and coding, and matching. Figure 2.2 shows the outlook of our FKP image acquisition device whose overall size is 160mm×125mm×100mm.

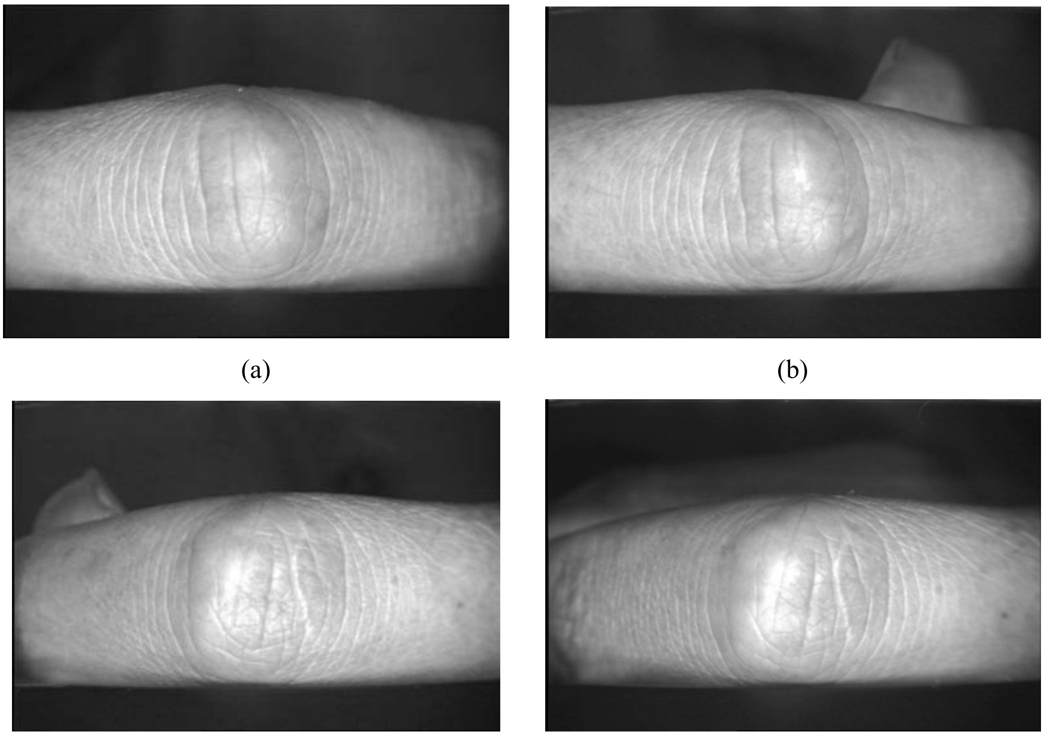
(a)



(b)

Figure 2.2 (a) The outlook of the developed FKP image acquisition device; (b) The device is being used to collect FKP samples.

A critical issue in data acquisition is to make the data collection environment as stable and consistent as possible so that variations among images collected from the same finger can be reduced to the minimum. In general, a stable image acquisition process can effectively reduce the complexity of the data processing algorithms and improve the image recognition accuracy. Meanwhile, we want to put as little constraint as possible on the users in order for high user friendliness of the system. With the above considerations, a semi-closed data collection environment is designed in our system. The LED light source and the CCD camera are enclosed in a box so that the illumination is nearly constant. One difficult problem is how to make the gesture of the finger be nearly constant so that the captured FKP images from the same finger are consistent. In our system, the finger bracket is designed for this purpose.



(a) (b) (c) (d)

Figure 2.3 Sample FKP images acquired by the developed system. (a) and (b) are from one finger while (c) and (d) are from another finger. Images from the same finger are taken at two different sessions with an interval of 56 days.

Refer to Figure 2.1, a basal block and a triangular block are used to fix the position of the finger joint. In data acquisition, the user can easily put his/her finger on the basal block with the middle phalanx and the proximal phalanx touching the two slopes of the triangular block. Such a design aims at reducing the spatial position variations of the finger in different capturing sessions. The triangular block is also used to constrain the angle between the proximal phalanx and the middle phalanx to a certain magnitude so that line features of the finger knuckle surface can be clearly imaged. The angle of the triangular block is 135°. That means when the FKP sample is being collected, the person's finger is bent with an angle about 135°.

After the image is captured, it is sent to the data processing module for preprocessing, feature extraction and matching (refer to Chapters 3, 4, and 5 for details). The size of the acquired FKP images is 768×576 under a resolution about 400 dpi. Figure 2.3 shows four sample images acquired by the developed device. Two images in the first row are from one finger and images in the second row are from another finger. Example images for the same finger were captured at two different collection sessions with an interval of 56 days. We see that by using the developed system, images from the same finger but collected at different times are very similar to each other. Meanwhile, images from different fingers are very different, which implies that FKP has the potential for personal identification.

2.2 Database Establishment

In order to evaluate the proposed FKP-based personal authentication system, an FKP database was established by using the developed FKP image acquisition system (refer to Figure 2.2). This database is intended to be a benchmark to evaluate the performance of various FKP recognition methods, and it is now publicly available at [109]. The FKP images were collected from 165 volunteers, including 125 males and 40 females. Among them, 143 subjects are 20~30 years old and the others are 30~50 years old. The volunteers were students and teachers from the Hong Kong Polytechnic University and Harbin Institute of Technology.

We collected the samples in two separate sessions. In each session, the subject was asked to provide 6 images for each of the left index finger, the left middle finger, the right index finger and the right middle finger. Therefore, 48 images from 4 fingers were collected from each subject. In total, the database contains 7,920 images from 660 different fingers. The average time interval between the first and the second sessions was about 25 days. The maximum and minimum time intervals were 96 days and 14 days respectively.

In all of the recognition experiments conducted in the following chapters, we took images collected at the first session as the gallery set and images collected at the second session as the probe set.

2.3 Data Preprocessing and ROI Extraction

2.3.1 Selection of the Image Resolution

Table 2.1 EERs obtained under different resolutions.

resolution	EER (%)
200 dpi	1.73
170 dpi	1.41
150 dpi	1.36
120 dpi	1.71
100 dpi	1.92

The resolution of original FKP images acquired in our system is about 400 dpi, which may not be optimal in terms of the accuracy and efficiency of FKP verification. In fact, many factors, such as the storage space, the computational cost, the employed feature extraction and matching method, and the recognition accuracy, should be considered in selecting a suitable resolution of the FKP images for a more efficient biometric system. To this end, we conducted a series of experiments to select the "optimal" resolution and set the selection criterion as: the minimum resolution with which a satisfying verification performance could be obtained. The verification performance was measured by the Equal Error Rate (EER), which is defined in section 1.2. The experiments were performed on a sub-dataset of the whole FKP database. In this sub-dataset, there were 120 classes, including 1,440 images. With respect to the feature extraction method, the CompCode, which will be introduced in Chapter 3, was used [51, 96]. We smoothed the original images by using a Gaussian kernel and then down-sampled the images to five lower resolutions: 200 dpi, 170 dpi, 150 dpi, 120 dpi and 100 dpi. The experimental results are summarized in Table 2.1.

Based on the results listed in Table 2.1, it can be seen that 150 dpi is a good choice. It leads to the lowest EER, while the resolution is much smaller than the original one (400 dpi). Such a downloading step will reduce the computational cost and speed up the feature extraction and matching processes significantly. Therefore, in all of the following experiments, we used the FKP images with a resolution 150 dpi.

2.3.2 ROI Extraction

FKP images collected from different fingers are very different. On the other hand, for the same finger, images collected at different collection sessions will also vary because of the variation of spatial locations of the finger. Therefore, it is necessary and critical to align FKP images by adaptively constructing a local coordinate system for each image. With such a coordinate system, an ROI can be cropped from the original image in order for reliable feature extraction and matching. The ROI extraction step can reduce the data amount in the feature extraction and matching stage and can reduce the influence of variations of the FKPs. In this sub-section, we will propose an algorithm for the local coordinate system determination and ROI sub-image extraction.

Because the finger is always put flatly on the basal block when the FKP image is captured, the bottom boundary of the finger is stable in every image and can be taken as the X-axis of the ROI coordinate system. However, the Y-axis is much more difficult to determine. Intuitively, we want to locate the Y-axis in the center of the phalangeal joint so that most of the useful features in the FKP image can be preserved within the ROI. It can be observed that line features on the two sides of the phalangeal joint have different convex directions. Taking this fact as a hint, we propose to code line pixels based on their convex directions and then make use of the convex direction codes to determine the Y-axis. Figure 2.4 illustrates the main steps of the coordinate system determination and the ROI extraction. In the following, we describe these steps in detail.

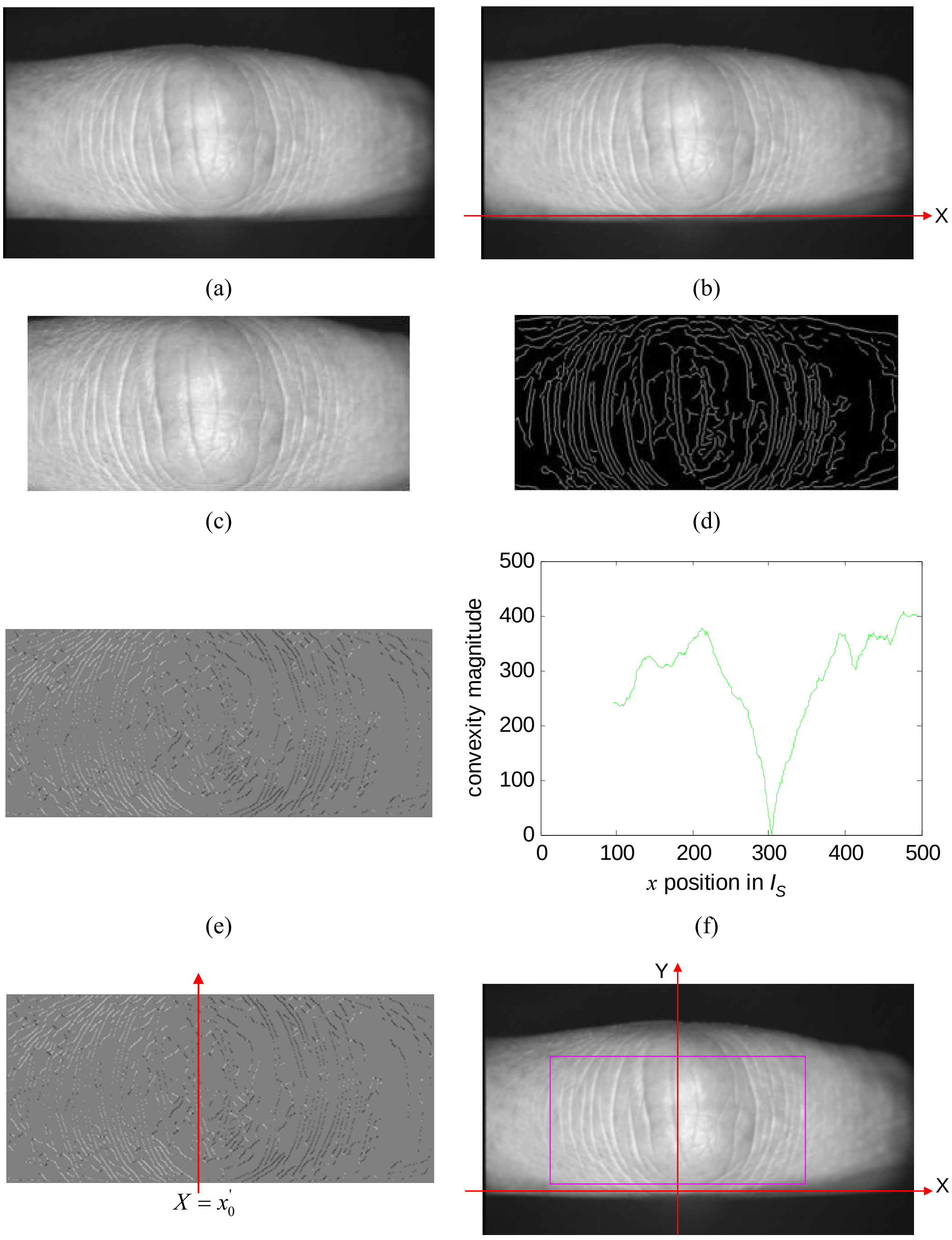


Figure 2.4 Illustration for the ROI extraction process. (a) I_D image which is obtained by a down-sampling operation after a Gaussian smoothing; (b) X-axis of the coordinate system, which is the line $Y = Y_0$, fitted from the bottom boundary of the finger; (c) I_S image extracted from I_D; (d) I_E image obtained by applying a Canny edge detector on I_S; (e) I_{CD} image obtained by applying the convex direction coding scheme to I_E; (f) plot of $conMag(x)$ for a typical FKP image; (g) line $X = x_0^{'}$, where $x_0^{'} = \arg\min_x (conMag(x))$; (h) ROI coordinate system, where the rectangle indicates the area of the ROI sub-image that will be extracted.

Step 1: image down-sampling

The size of the captured FKP image is 768×576 under a resolution of 400dpi. Based on our experiments, it is not necessary to use such a high resolution for feature extraction and matching. Therefore, we apply a Gaussian smoothing operation to the original image, and then down-sample the smoothed image to about 150 dpi (see Section 2.3.1 for the discussion of resolution selection). The down-sampling operation has two advantages. First it can significantly reduce the computational cost by reducing the data amount. Second, the Gaussian smoothing will suppress the noise in the original image, which can benefit the following feature extraction and matching steps. We denote by I_D the down-sampled image and Figure 2.4(a) shows such an image.

Step 2: determine the *X*-axis of the coordinate system

Refer to Figure 2.4(b), the bottom boundary of the finger can be easily extracted by a Canny edge detector. Actually, this bottom boundary is nearly consistent to all FKP images because all the fingers are put flatly on the basal block in data acquisition. By fitting this boundary as a straight line, the *X*-axis of the local coordinate system is determined.

Step 3: crop a sub-image I_S from I_D

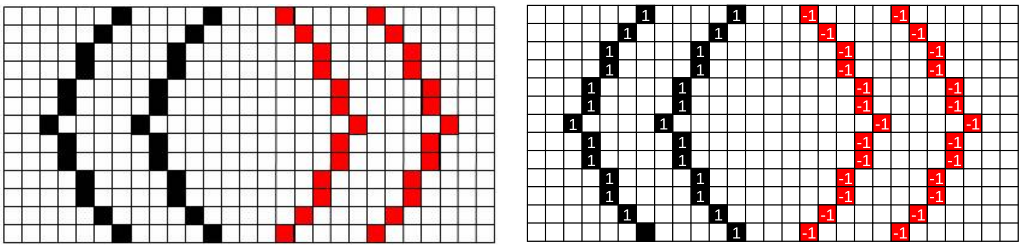
Useful information which can be used for personal identification only resides in a portion of the whole FKP image. Therefore, we first crop a sub-image I_S from the original image for the convenience of later processing. The left and right boundaries of I_S are two fixed values evaluated empirically. The top and bottom boundaries are estimated according to the boundary of real fingers. Figure 2.4(c) shows an example I_S image. This roughly cropped sub-image will be used to calculate the *Y*-axis so that an accurate ROI image can be cropped.

Step 4: Canny edge detection

By applying Canny edge detector to I_S, the corresponding edge map I_E can be obtained. See

Figure 2.4(d) for an example.

Step 5: convex direction coding for I_E



(a) (b)

Figure 2.5 (a) Ideal model for FKP "curves"; (b) Convex direction coding scheme.

Table 2.2 Algorithm Convex_Direction_Coding (I_E)

Input: I_E ($m \times n$ binary edge map computed in step 4)
Output: I_{CD} ($m \times n$ convex direction code map)
Begin module
$y_{mid} = \frac{height\ of\ I_E}{2}$;
for each $I_E(i,j)$ **do**
if $I_E(i,j) = 0$ *// it is a background pixel*
$I_{CD}(i,j) = 0$;
else if $I_E(i+1, j-1) = 1\ and\ I_E(i+1, j+1) = 1$ *// it is a bifurcation pixel*
$I_{CD}(i,j) = 0$;
else if $(I_E(i+1, j-1) = 1\ and\ i <= y_{mid})\ or\ (I_E(i+1, j+1) = 1\ and\ i > y_{mid})$
$I_{CD}(i,j) = 1$;
else if $(I_E(i+1, j+1) = 1\ and\ i <= y_{mid})\ or\ (I_E(i+1, j-1) = 1\ and\ i > y_{mid})$
$I_{CD}(i,j) = -1$;
end if
end for
End module

Based on the local convexity characteristics of the edge map I_E, we can code I_E to get the convex direction coding map I_{CD}. At this step, each pixel on I_E will be given a code to represent the local convex direction of this pixel. The underlying principle of this coding scheme is as follows. Based on the observation of FKP images, we abstract an ideal model for "curves" on an FKP image as shown in Figure 2.5(a). In this model, an FKP "curve" is either convex leftward or convex rightward. We code the pixels on convex leftward curves as "1", the pixels on convex rightward curves as "-1", and the other pixels not on any curves as "0". Figure 2.5(b) illustrates the coding scheme. In our system, we regard the edges obtained in step 4 as "curves" and this convex direction coding is performed on I_E. The pseudo code of this algorithm is given in Table 2.2:

After convex direction coding, each I_{CD} point is assigned a value 0, 1 or -1. Figure 2.4(e) shows an I_{CD} map in false color image format. White pixels on it are the ones with convexity value "1"; black pixels are the ones with value "-1"; and gray pixels are of value "0".

Step 6: determine the *Y*-axis of the coordinate system

Consider the ideal FKP curve model set up at step 5. For an FKP image, "curves" on the left part of phalangeal joint are mostly convex leftward and those on the right part are mostly convex rightward. Meanwhile, "curves" in a small area around the phalangeal joint do not have obvious convex directions. Based on this observation, at a horizontal position *x* (*x* represents the column) of an FKP image, we define the "convexity magnitude" as:

$$conMag(x) = abs\left(\sum_{W} I_{CD}\right) \tag{2-1}$$

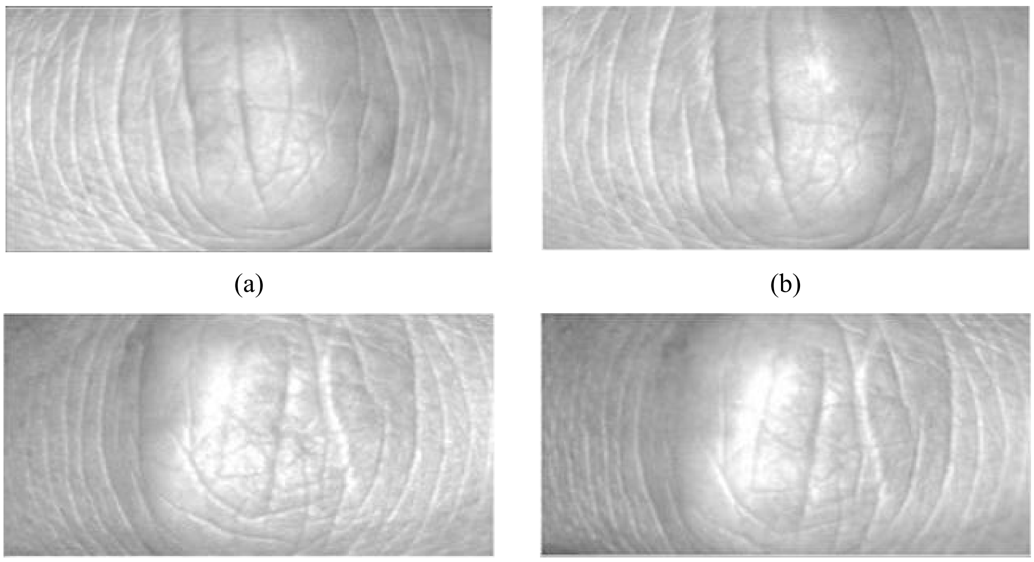
where W is a window that is symmetrical about the axis $X = x$ and W is of size $d \times h$ with h being the height of I_S. d is experimentally chosen as 35 in this thesis. The "convexity magnitude" is proposed to measure how strong the dominant convex direction is in a local area on the FKP image. The characteristic of the FKP image suggests that $conMag(x)$ will reach a minimum around the center of the phalangeal joint and this position can be used to set the *Y*-axis of the coordinate system. Let

$$x_0^{'} = \arg\min_{x}\left(conMag(x)\right) \tag{2-2}$$

Then $X = x_0^{'}$ can be set as the *Y*-axis. Figure 2.4(f) plots the curve *conMag*(*x*) of an FKP image and Figure 2.4(g) shows the vertical line $X = x_0^{'}$, which is the *Y*-axis of the ROI system.

Step 7: crop the ROI image

Now that we have fixed the *X*-axis and *Y*-axis, the local coordinate system can then be determined. Refer to Figure 2.4(h), with the constructed coordinate system, the ROI sub-image I_{ROI} can be extracted from I_D with a fixed size, which is empirically set as 110×220 in our system.



(a) (b) (c) (d)

Figure 2.6 Sample ROI images extracted by the proposed method. These four images are ROI images for the sample images shown in Figure 2.3, respectively.

Figure 2.6 shows some examples of the extracted ROI images. We can see that the proposed coordinate system construction and ROI extraction method can effectively align the different FKP images and normalize the area for feature extraction. Such operations reduce greatly the variations caused by the various poses of the finger in data collection. It needs to be noted that ROI images extracted from our collected FKP samples using the proposed algorithm are also publicly available at [109].

2.4 Summary

In this chapter, the overall architecture of the proposed FKP recognition system, the imaging device, the collected database, the data preprocessing, and the ROI extraction are presented. The design of our FKP imaging device has the following merits: 1) the acquisition device could be easily made to a small size; 2) image around the finger knuckle area is captured directly, which largely simplifies the following data preprocessing steps; and 3) since the finger knuckle is slightly bent when being captured, the distinctive FKP texture patterns can be clearly imaged, which makes the proposed FKP system have high accuracy. The collected FKP database has been made publicly available in order to advance the research of the biometric community.

In the following Chapters 3~5, we will focus on the feature extraction and matching of FKP images.

Chapter 3. Feature Extraction and Matching I: Coding-based Methods

As in any pattern classification task, the feature extraction and matching plays a key role in our FKP-based personal authentication system. Chapters 3~5 of this thesis will focus on designing appropriate feature extraction and matching algorithms for FKP images matching. This chapter will focus on coding-based methods. After reviewing three state-of-the-art coding based schemes, we present a new coding-based feature extraction and matching method, namely ImCompCode&MagCode [98], which is actually an extension for CompCode [51, 96]. In addition, we present other two efficient and compact coding approaches, RCode1 and RCode2, based on Riesz transforms [105]. All the coding methods mentioned in this chapter will be thoroughly examined in the experiments part.

3.1 State-of-the-art Coding Methods

In the past decade, an enormous volume of literature has been devoted to investigate various feature extraction methods for matching biometric images full of line-like structures, such as palmprint and Iris. Among them, coding-based methods have been widely and successfully used. The prevalence of coding-based methods should be attributed to the great success of the IrisCode invented by Daugman for iris recognition [23]. Compared with the other methods, coding-based methods have many advantages. Generally speaking, they have the merits of high accuracy, robustness to illumination variation and fast feature extraction and matching speed. Daugman has demonstrated that the real-time brute-force identification in large databases is possible with coding based features and the associated bit-wise Hamming distance based matching. In a typical coding-based method, each field of the code map is assigned a bit-wised code, based on the quantization of the image responses to a set of filters. Three state-of-the-art coding methods, CompCode [51], RLOC [59], and OrdinalCode [53], are reviewed in this sub-section.

3.1.1 Competitive Code (CompCode)

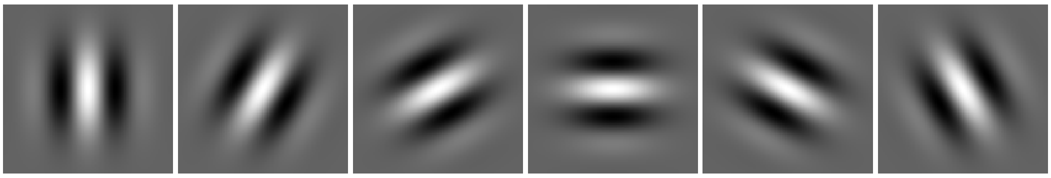


Figure 3.1 Real parts of six Gabor filters with different orientations.

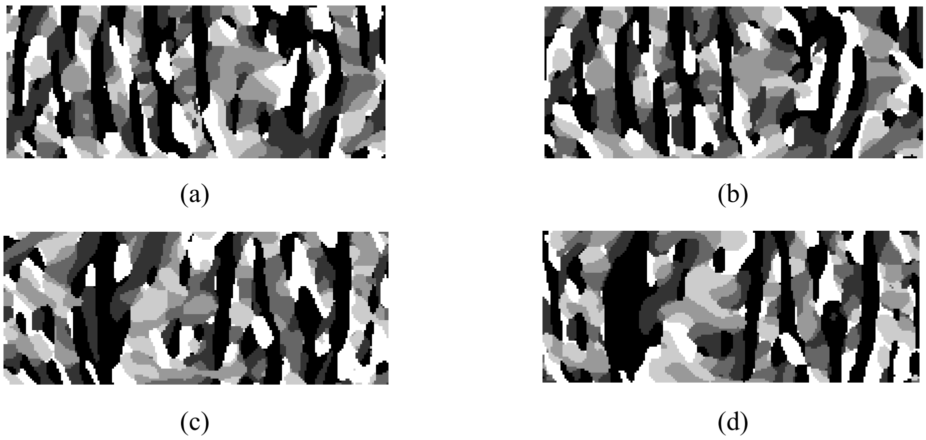


Figure 3.2 (a) ~ (d) CompCode maps for the FKP ROI images shown in Figure 2.6, respectively.

CompCode is based on the image's response to a set of Gabor filters [107], which are defined as

$$G(x,y) = \exp\left(-\frac{1}{2}\left(\frac{x'^2}{\sigma_x^2} + \frac{y'^2}{\sigma_y^2}\right)\right) \cdot \exp\left(i2\pi f x'\right) \quad (3\text{-}1)$$

where $x' = x\cos\theta + y\sin\theta$, $y' = -x\sin\theta + y\cos\theta$. In Eq. (3-1), f represents the frequency of the sinusoid factor, θ represents the orientation of the normal to the parallel stripes of the Gabor function, σ_x and σ_y are the standard deviations of the 2D Gaussian envelop. Gabor filter has been widely used as an effective tool to fulfill the feature extraction tasks in many biometrics systems, such as face [14, 18, 19, 21, 22], iris [23-25], fingerprint [38], palmprint [51, 54, 57], etc. The frequency and orientation representations of Gabor filters are similar to those of the human visual system [110, 111]. Three kinds of features, orientation, phase, and magnitude can be extracted with Gabor filters [112].

Denote by G_R the real part of the Gabor filter G. With six G_Rs sharing the same parameters, except the parameter of orientation, as shown in Figure 3.1, the local orientation information of the image I at the position (x, y) can be extracted and coded. Mathematically, this competitive coding process can be expressed as

$$CompCode(x,y) = \arg\min_j \{I(x,y) * G_R(x,y,\theta_j)\} \tag{3-2}$$

where * stands for the convolution operation and $\theta_j = j\pi/6$, $j = \{0,\ldots, 5\}$. Then, the dominant orientation $\{0, \pi/6, \pi/3, \pi/2, 2\pi/3, 5\pi/6\}$ will be encoded with three bits as {000, 001, 011, 111, 110, 100} for efficient representation and bitwise matching. It can be seen that six convolutions with the image are needed to extract its CompCode. Example CompCode maps extracted from FKP images are shown in Figure 3.2.

3.1.2 Robust Line Orientation Code (RLOC)

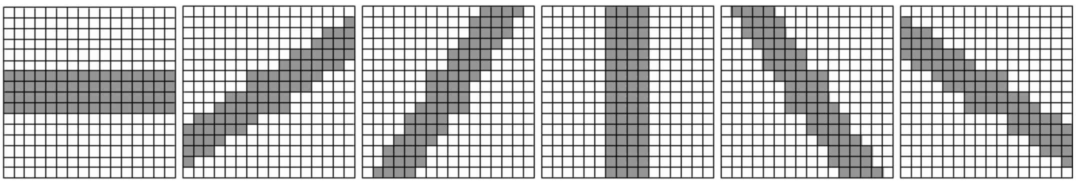
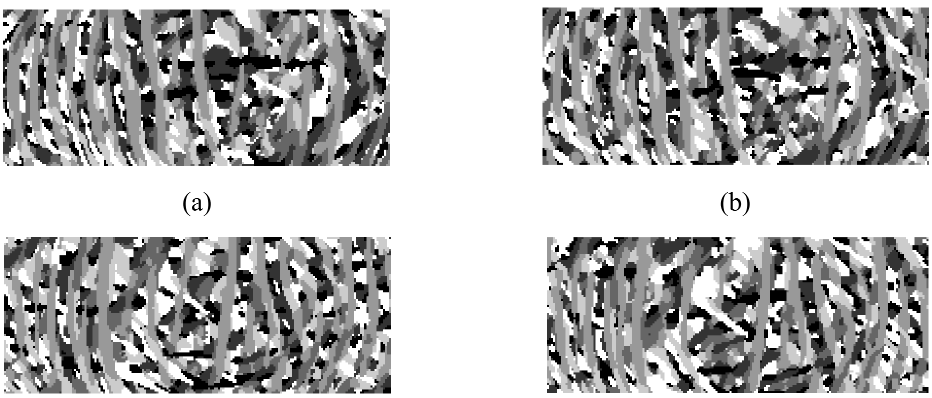


Figure 3.3 Six binary line templates used in RLOC. "Gray" fields are of value 1 while "white" fields are of value 0.



(a) (b) (c) (d)

Figure 3.4 (a) ~ (d) RLOC maps for the FKP ROI images shown in Figure 2.6, respectively.

Robust line orientation code (RLOC) proposed by Jia *et al*. [59] is another effective method to encode the local orientation information of an image. It assumes that lines in the examined image are negative lines. For palmprint line orientation estimation, Jia *et al*. [59] devised six binary line templates, $T_0 \sim T_5$, with the line width lw = 4 pixels, as shown in Figure 3.3. These six line templates are of the orientations 0, $\pi/6$, $\pi/3$, $\pi/2$, $2\pi/3$, and $5\pi/6$. In Figure 3.3, “gray” fields are of value 1 while “white” fields are of value 0. Then using the “winner-take-all” rule, RLOC is defined as

$$RLOC(x,y) = \arg\min_j \{I(x,y) * T_j(x,y)\},\ j \in \{0,...,5\} \tag{3-3}$$

Then, the extracted orientation can be coded with 3-bits using the same scheme as described in CompCode. Example RLOC maps extracted from FKP images are shown in Figure 3.4.

3.1.3 Ordinal Code

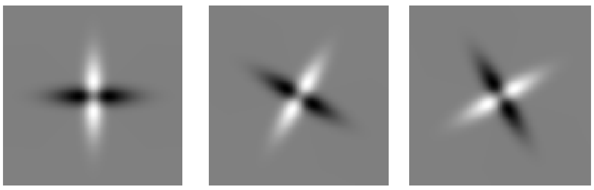
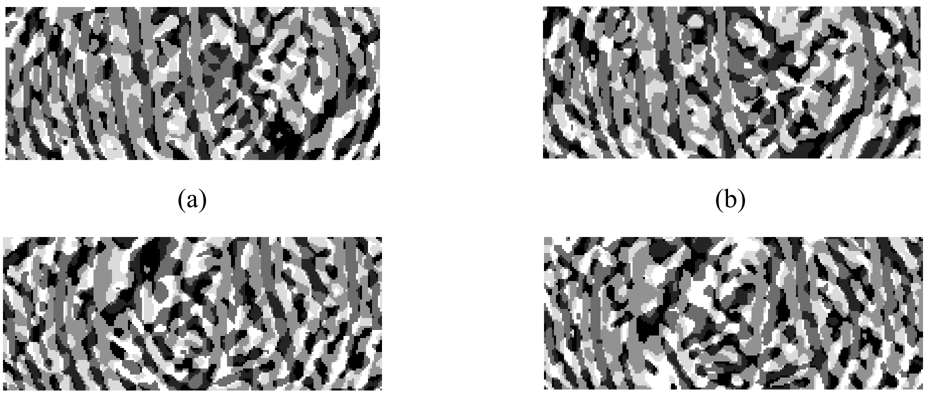


Figure 3.5 Three ordinal filters $OF(0)$, $OF(\pi/6)$, and $OF(\pi/3)$ used in OrdinalCode.



(a) (b)

(c) (d)

Figure 3.6 (a) ~ (d) OrdinalCode maps for the FKP ROI images shown in Figure 2.6, respectively.

In OridinalCode [53], the 2D Gaussian filter is used to calculate the weighted average intensity of a line-like region. It can be expressed as

$$Gau(x,y,\theta)=\exp\left(-\frac{\left(x\cos\theta+y\sin\theta\right)^2}{s_x^2}-\frac{\left(-x\sin\theta+y\cos\theta\right)^2}{s_y^2}\right) \quad (3\text{-}4)$$

where θ denotes the dominant orientation of the 2D Gaussian filter, and s_x and s_y are the two standard deviations of the 2D Gaussian filter.

Based on the 2D Gaussian filter, the ordinal filter, which can compare two orthogonal line-like image regions, is designed as

$$OF(x,y,\theta)=Gau(x,y,\theta)-Gau(x,y,\theta+\pi/2) \quad (3\text{-}5)$$

For each local region in the examined image, three ordinal filters $OF(0)$, $OF(\pi/6)$, and $OF(\pi/3)$, as shown in Figure 3.5, are applied on it to obtain 3-bits ordinal codes based on the sign of the filtering results. Example OrdinalCode maps extracted from FKP images are shown in Figure 3.6.

3.1.4 Matching

As stated, given an input image, the three coding schemes CompCode, RLOC, and OrdinalCode will have similar outputs in form, i.e., code maps consisting of three bit-planes, although the filters and the coding strategies adopted are different. Therefore, they can use a universal matching scheme, namely the normalized Hamming distance, to measure the dissimilarity of the two given code maps. Suppose that P and Q are the two code maps extracted by any coding scheme described above. Then, their normalized Hamming distance is defined as

$$d(P,Q)=\frac{\sum_{y=1}^{Rows}\sum_{x=1}^{Cols}\sum_{i=0}^{2}\left(P_i^b(x,y)\otimes Q_i^b(x,y)\right)}{3S} \quad (3\text{-}6)$$

where $P_i^b(Q_i^b)$ is the i^{th} bit plane of P (Q), S is the area of the code map, and $\otimes$ represents the bitwise "exclusive OR" operation.

In practice, taking into account the possible translations in the extracted ROI sub-image with respect to the one extracted in the enrolment, multiple matches are performed by translating one set of features in horizontal and vertical directions. And the minimum of the resulting matching distances is considered to be the final matching distance. In such cases, *S* is the area of the overlapping parts of the two code maps.

3.2 ImCompCode&MagCode

In this section, we will present a new coding-based feature extraction and matching method for FKP recognition. It is actually an extension of the CompCode method. Specifically, using Gabor filtering, we propose an improved competitive coding (ImCompCode) method to exploit the orientation information, and then we propose a magnitude coding (MagCode) method to exploit magnitude information. Finally, we fuse these two kinds of features in FKP matching.

3.2.1 Improved Competitive Coding (ImCompCode) for Orientation Extraction

Here we improve the original CompCode scheme. Often on an FKP image, there are some pixels lying on relatively "plane" areas, i.e. these pixels do not reside on any lines and consequently do not have a dominate orientation. Accordingly, the *J* Gabor filter responses at such pixels do not have much variation. If we still assign an orientation code to it, this code may not be stable and will be sensitive to noise, making the performance of feature representation and matching decreased. Therefore, those "plane" pixels should be removed from orientation coding. We define the "orientation magnitude" at a pixel as:

$$oriMag(x,y)=\frac{abs\left(\max(R)-\min(R)\right)}{\max\left(abs\left(\max(R)\right),abs\left(\min(R)\right)\right)} \tag{3-7}$$

where $R = \{R_j = I(x, y) * G_R(x, y, \theta_j)\}$, $j = \{0,\ldots, J - 1\}$ are the Gabor filtering responses at this pixel. The "orientation magnitude" $oriMag(x, y)$ can measure how likely the pixel

(x, y) has a dominant orientation. If it is smaller than a threshold, we reckon that this pixel has no dominant orientation and the corresponding competitive code is assigned as J. Based on the experimental results, using 6 Gabor filters of different orientations are enough. This is in accordance with the conclusion made by Lee [113] that simple neural cells are sensitive to specific orientations with approximate bandwidths of $\pi/6$. The pseudo code for our ImCompCode scheme is summarized in Table 3.1 and Figure 3.7(a)~(d) are ImCompCode maps for FKP ROI images shown in Figure 2.6.

Table 3.1 Algorithm ImCompCode (I_{ROI})

Input: I_{ROI} ($m \times n$ ROI sub-image)

Output: $ImCompCode$ ($m \times n$ integer matrix)

Begin module

for each $I_{ROI}(x,y)$ **do**

$$R = \left\{ R_j = I_{ROI}(x,y) * G_R(x,y,\theta_j) \right\}, \text{ where } \theta_j = j\pi / 6, j = \{0,1,...5\}$$

$$oriMag = \frac{abs\left(\max(R) - \min(R)\right)}{\max\left(abs\left(\max(R)\right), abs\left(\min(R)\right)\right)}$$

if $oriMag < T$ *// this pixel does not have a dominant orientation*

$ImCompCode(x,y) = 6$;

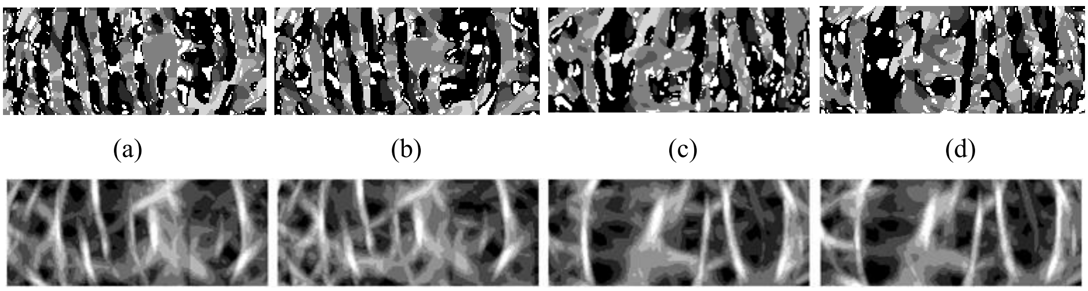
else

$$ImCompCode(x,y) = \arg\min_j \left\{ R_j \right\}$$

end if

end for

End module



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 3.7 (a) ~ (d) and (e) ~ (h) are ImCompCode maps and MagCode maps for the FKP ROI images shown in Figure 2.6, respectively.

3.2.2 Magnitude Coding (MagCode) for Magnitude Extraction

Besides orientation information, we also want to exploit magnitude information from Gabor filter responses. The magnitude of the Gabor filter response at $I(x, y)$ is:

$$\sqrt{\left(I(x,y)*G_R(x,y,\theta_j)\right)^2+\left(I(x,y)*G_I(x,y,\theta_j)\right)^2} \quad (3\text{-}8)$$

where G_R and G_I represent the real part and the imaginary part of the Gabor function G respectively. However, in order to reduce the computational cost, when generating the magnitude code map, we want to make use of the temporary results generated from the "orientation coding" process. Thus, we still only use the real part of the Gabor filter and define the magnitude at $I(x, y)$ as:

$$mag(x,y)=\max_j\left(abs\left(I(x,y)*G_R(x,y,\theta_j)\right)\right) \quad (3\text{-}9)$$

Then a localized quantization is applied to $mag(x, y)$ to get the magnitude code. This process can be expressed as:

$$magCode(x,y)=ceil\left(\left(mag(x,y)-lmin\right)\Big/\left(\frac{lmax-lmin}{N}\right)\right) \quad (3\text{-}10)$$

where N is the number of quantization levels, $lmin=\min_{(x,y)\in W_m}\left(mag(x,y)\right)$, and $lmax=\max_{(x,y)\in W_m}\left(mag(x,y)\right)$. W_m is a $w\times w$ window centered at (x, y). The resulting magnitude code is an integer within 1~ N. w and N can be tuned by experiments on a sub-dataset and they are experimentally set as 31 and 8 in this thesis, respectively. Figure 3.7(e) ~ (h) show magnitude code maps for FKP ROI images presented in Figure 2.6.

3.2.3 Matching

Suppose we are comparing two FKP ROI images, P and Q. Let P_o and Q_o be the two orientation code maps; and let P_m and Q_m be the two magnitude code maps. At first, we will calculate the matching distance between P_o and Q_o and the matching distance between P_m and Q_m respectively, and then fuse the two matching distances together as the final matching distance between P and Q.

When calculating the matching distance between P_o and Q_o, we adopt the angular distance proposed in [51], which is defined as:

$$angD(P,Q)=\frac{\sum_{y=1}^{Rows}\sum_{x=1}^{Cols}\varphi\left(P_o(x,y),Q_o(x,y)\right)}{(J/2)\cdot S} \tag{3-11}$$

where S is the area of the code map, and

$$\varphi\left(P_o(x,y),Q_o(x,y)\right)=\begin{cases}1, P_o(x,y)=6 \; and \; Q_o(x,y)\neq 6\\ 1, P_o(x,y)\neq 6 \; and \; Q_o(x,y)=6\\ 0, P_o(x,y)=Q_o(x,y)\\ \min\left(P_o(x,y)-Q_o(x,y), Q_o(x,y)-\left(P_o(x,y)-6\right)\right),\\ \;if \; P_o(x,y)>Q_o(x,y) \; and \; P_o(x,y)\neq 6\\ \min\left(Q_o(x,y)-P_o(x,y), P_o(x,y)-\left(Q_o(x,y)-6\right)\right),\\ if \; P_o(x,y)<Q_o(x,y) \; and \; Q_o(x,y)\neq 6\end{cases} \tag{3-12}$$

The matching distance between P_m and Q_m is defined as:

$$magD(P,Q)=\frac{\sum_{y=1}^{Rows}\sum_{x=1}^{Cols}abs\left(P_m(x,y)-Q_m(x,y)\right)}{(N-1)\cdot S} \tag{3-13}$$

Then, the final matching distance between P and Q can be fused from *angD* and *magD* as:

$$dist(P,Q)=(1-\lambda)\cdot angD(P,Q)+\lambda\cdot magD(P,Q) \tag{3-14}$$

where λ is used to control the contribution of *magD* to *dist* and it is experimentally set as 0.15 in this thesis.

3.3 Riesz Transforms based Coding Methods

Several state-of-the-art coding based feature extraction and matching methods have been reviewed and one novel coding method has been presented in the previous sections. Undoubtedly, the core of a coding-based method is how to devise a “perfect” coding scheme. These factors should be carefully considered when devising a coding scheme: the computation complexity, the compactness (number of bits for each code), the robustness to the illumination changes, and the distinctiveness. Based on the recent studies in the signal/image processing community, the local image information can be well characterized in a unified theoretic framework, namely Riesz transform [105], which actually is the vector-valued extension of the Hilbert transform. Recently, Riesz transforms have attracted much attention in the signal/image processing community [100-103, 114-131]. Felsberg and Sommer are the first to bring the Reisz transform to the signal/image processing community [114]. In their work, they proposed the monogenic signal based on the 1st-order Riesz transforms. The monogenic signal can be regarded as a 2D extension to the classical analytic signal. Using the monogenic signal, the local phase and the local orientation of the intrinsically 1D signal can be extracted and represented in a compact and isotropic way. The monogenic signal has already been used in some image-processing applications, such as the local structure analysis [115, 118], the stereo motion estimation [126], the image registration [125], the optical flow estimation [117], the image quality assessment [101], the texture recognition [102], and the face recognition [103], etc. However, from the theoretical view, 1st-order Riesz transform based monogenic signal is designed only for intrinsically 1D signal. In order to characterize the intrinsically 2D local structures, higher order Riesz transforms are needed. In [120, 124], Wietzke and Sommer used up to 3rd-order Riesz transforms to construct the signal multi-vector which allows modeling any local image structure such as lines, edges, corners, and junctions in scale space in one unified framework. Since using Riesz transforms local image patterns can be well represented in a unified framework, in this sub-section we propose to quantify the image's responses to the Riesz transforms to devise new coding schemes. Specifically, two different Riesz transforms based coding schemes are proposed, namely RCode1 and RCode2. They both use 3-bits to represent each code and resort to the normalized Hamming distance for matching. Their performances for FKP recognition will also be evaluated.

3.3.1 Fundamentals of Riesz Transforms

Before we present our Riesz transforms based coding approaches, at first we need to give a brief review of the Riesz transforms since it is not well-known in the engineering community. The mathematical fundamentals are partly based on the authoritative book of Stein [105] and partly on the extensive works of the Cognitive Systems Group in Kiel University [114-124].

A. Hilbert Transform and the Analytic Signal

For $f \in L^p(R)$, $1 \leqslant p < \infty$, the Hilbert transform of f, Hf, is defined as [105, 132]

$$(Hf)(x) = \frac{1}{\pi} p.v.\int_{\mathbb{R}} \frac{f(t)}{x-t} dt \tag{3-15}$$

where *p.v.* stands for the Cauchy principal value. Equally, H can be expressed by the convolution kernel in the spatial domain as $h(x) = 1/\pi x$. The Fourier transform of the kernel h is $\hat{h}(u) = -ju/|u|$, where $j^2 = -1$.

The Hilbert transform based analytic signal, first proposed by Denis Gabor [103], is a powerful tool for the 1D signal analysis. Given a 1D real signal $f(x)$, the corresponding analytic signal is defined as

$$f_a(x) = f(x) + jHf(x) = A(x)e^{j\phi(x)} \tag{3-16}$$

where $A(x) = \sqrt{(f(x))^2 + (Hf(x))^2}$ is the local amplitude while $\phi(x) = \arctan 2(Hf(x), f(x)) \in [0, 2\pi)$ is the instantaneous phase. The advantage of this representation is that it allows one to obtain the time-varying amplitude and phase of a 1D signal. After its birth, the analytic signal has been utilized in various applications involving some kind of amplitude or frequency modulation [132, 133].

B. 1st-order Riesz Transform and the Monogenic Signal

In the literature, there are many attempts reported to generalize the analytic signal to 2D and among them, the monogenic signal [114] proposed by Felsberg and Sommer is the most distinguished one. The monogenic signal is built upon the 1^{st}-order Riesz transform which is a vector-valued extension of the Hilbert transform [105]. The convolution kernels of the Riesz transform in the nD spatial domain can be expressed as

$$R_j(\mathbf{y}) = c_n \frac{\mathbf{y}_j}{|\mathbf{y}|^{n+1}} \tag{3-17}$$

where $c_n = \Gamma[(n+1)/2]/\pi^{(n+1)/2}$, $\mathbf{y} = (y_1, y_2, \dots, y_n)$ and $j = 1, 2, \dots, n$. $\Gamma(x)$ is the Gamma function defined as [134]

$$\Gamma(\alpha) = \int_0^\infty x^{\alpha-1} e^{-x} dx, \alpha > 0 \tag{3-18}$$

Gamma function has the following properties

$$\Gamma(1) = 1 \tag{3-19}$$

$$\Gamma(\alpha+1) = \alpha\Gamma(\alpha) \tag{3-20}$$

$$\Gamma(n) = (n-1)!\text{, if } n \text{ is a positive integer} \tag{3-21}$$

$$\Gamma\left(\frac{1}{2}\right) = \sqrt{\pi} \tag{3-22}$$

In 2D, which is the case of interest for image processing applications (in this case, $n = 2$, $j = \{1, 2\}$, and $c_n = 1/2\pi$), the Riesz transform consists of two kernels expressed as

$$h_x(\mathbf{x}) = \frac{x}{2\pi|\mathbf{x}|^3}, h_y(\mathbf{x}) = \frac{y}{2\pi|\mathbf{x}|^3} \tag{3-23}$$

where $\mathbf{x} = (x, y) \in \mathrm{R}^2$. The Fourier transforms of h_x and h_y are

$$H_u(\mathbf{u}) = -j\frac{u}{|\mathbf{u}|}, H_v(\mathbf{u}) = -j\frac{v}{|\mathbf{u}|} \tag{3-24}$$

where $\mathbf{u} = (u, v) \in \mathrm{R}^2$. The interpretation of the Riesz transform can be performed in the Radon domain. The relationship among the Riesz transform, the classical Hilbert transform, and the Radon transform can be described as Figure 3.8. For more details about the physical interpretation of the Riesz transform, the reader can refer to [114, 120].

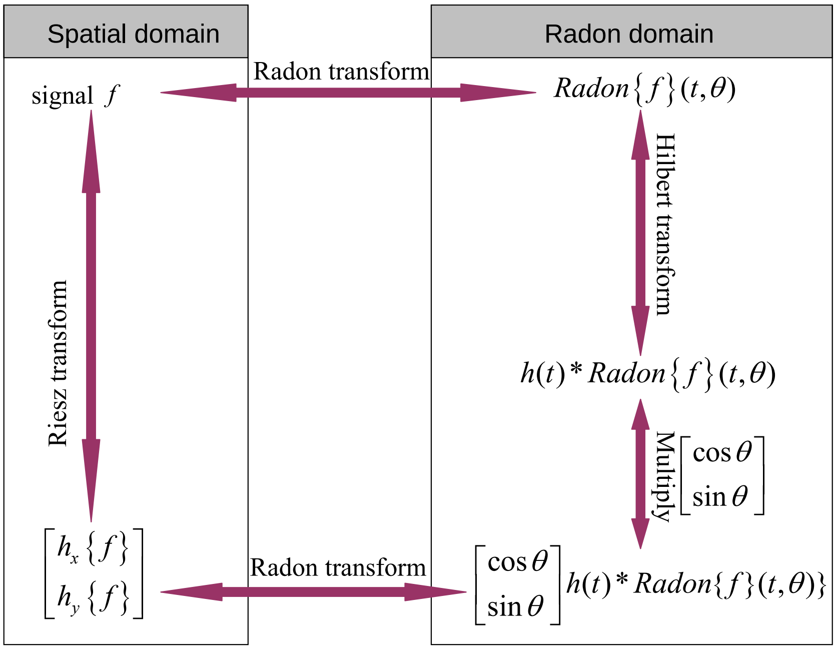
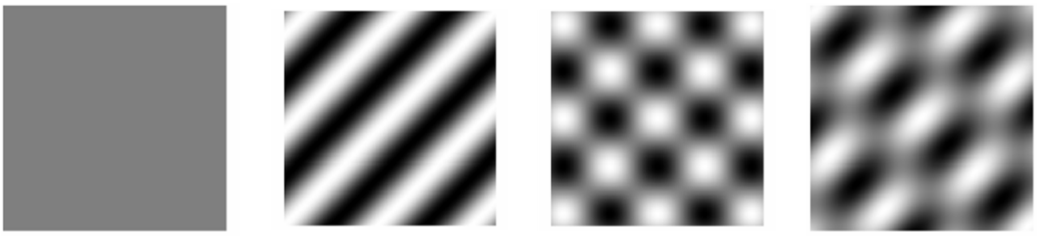


Figure 3.8 The relationship among the Riesz transform, the classical Hilbert transform, and the Radon transform.

Given a 2D signal $f(\mathbf{x})$, its corresponding monogenic signal $f_M(\mathbf{x})$ is defined as the combination of the original signal itself and its two Riesz transforms

$$f_M(\mathbf{x}) = \left(f(\mathbf{x}), h_x\{f\}(\mathbf{x}), h_y\{f\}(\mathbf{x})\right) \tag{3-25}$$

where $h_x\{f\}$ means convolving f with h_x. As the Riesz transform with respect to the Hilbert transform, the monogenic signal is a multi-dimensional isotropic generalization of the 1D analytic signal.



(a) (b) (c) (d)

Figure 3.9 Signals with different intrinsic dimensions. (a) i0D; (b) i1D; (c) i2D; (d) i2D.

To ease the following discussions, we introduce another concept here, the intrinsic dimension. The intrinsic dimension is the number of degrees of freedom necessary to describe a local image structure [135]. 2D image signals can be classified into local regions

N of different intrinsic dimensions. For example, constant areas are of intrinsic dimension zero (i0D) while straight lines and edges are of intrinsic dimension one (i1D). Mathematically, such a classification can be expressed as

$$f \in \begin{cases} \mathrm{i0}D_N, f(\mathbf{x}_i) = f(\mathbf{x}_j), \forall \mathbf{x}_i, \mathbf{x}_j \in N \\ \mathrm{i1}D_N, f(x,y) = g(x\cos\theta + y\sin\theta), \forall (x,y) \in N, f \notin \mathrm{i0}D_N \\ \mathrm{i2}D_N, else \end{cases} \tag{3-26}$$

where g is a 1D real-valued function. Examples of i0D, i1D, and i2D signals are shown in Figure 3.9.

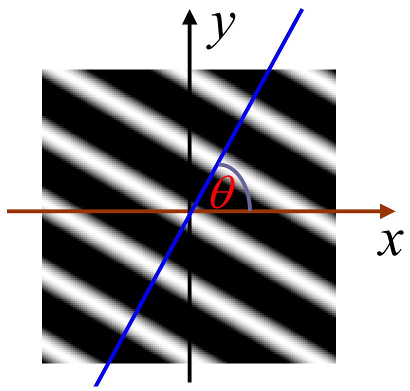


Figure 3.10 An example of the i1D signal $f_{\mathrm{i}1D}(x, y) = g(x\cos\theta + y\sin\theta)$, where g=cos(0.2t) and $\theta = \pi/3$.

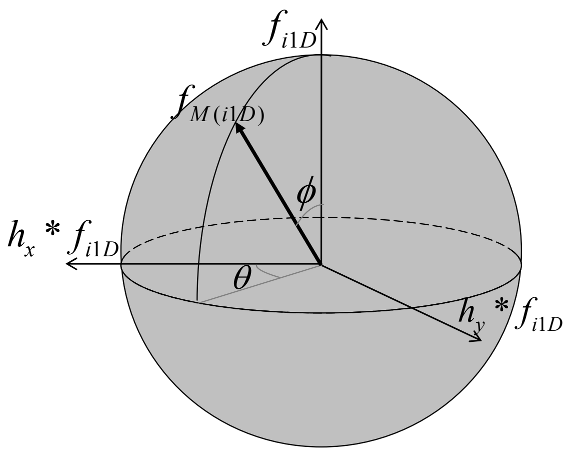


Figure 3.11 Geometric illustration of the monogenic signal in a spherical coordinate system.

Table 3.2 Conceptual comparison between the analytic signal and the monogenic signal.

	1D signal		2D signal	
	Concept	Formula	Concept	Formula
(hyper) complex signal	analytic signal	$f_a(x) = f(x) + jHf(x)$	monogenic signal	$f_M(\mathbf{x}) = \left(f(\mathbf{x}), h_x\{f\}(\mathbf{x}), h_y\{f\}(\mathbf{x})\right)$
transform	Hilbert	$f_H(x) = p.v.\{h(x) * f(x)\}$	Riesz	$f_{R_1}(\mathbf{x}) = p.v.\{h_x(\mathbf{x}) * f(\mathbf{x})\}$ $f_{R_2}(\mathbf{x}) = p.v.\{h_y(\mathbf{x}) * f(\mathbf{x})\}$
kernel		$h(x) = \frac{1}{\pi x}$		$h_x(\mathbf{x}) = \frac{x}{2\pi\lvert\mathbf{x}\rvert^3}$ $h_y(\mathbf{x}) = \frac{y}{2\pi\lvert\mathbf{x}\rvert^3}$
Fourier multiplier		$\hat{h}(u) = -j\frac{u}{\lvert u\rvert}$		$H_u(\mathbf{u}) = -j\frac{u}{\lvert\mathbf{u}\rvert}$ $H_v(\mathbf{u}) = -j\frac{v}{\lvert\mathbf{u}\rvert}$
local amplitude		$\lvert f_a(x)\rvert = \sqrt{(f(x))^2 + (Hf(x))^2}$		$\lvert f_M(\mathbf{x})\rvert = \sqrt{f^2(\mathbf{x}) + h_x^2\{f\}(\mathbf{x}) + h_y^2\{f\}(\mathbf{x})}$
argument (1D) zenith (2D)	Local phase	$\phi(x) = \arg\left(f(x) + jf_H(x)\right)$	local 1D phase	$\phi = \arctan 2\left(\operatorname{sgn}(h_x\{f\})\sqrt{h_x^2\{f\} + h_y^2\{f\}}, f\right)$
azimuth (2D)			local orientation	$\theta(\mathbf{x}) = \arctan\frac{h_y\{f(\mathbf{x})\}}{h_x\{f(\mathbf{x})\}}$

Consider an i1D signal $f_{i1D}(x, y) = g(x\cos\theta + y\sin\theta)$, where θ is its main orientation, as shown in Figure 3.10. For any point (x, y) on $f_{i1D}(x, y)$, denote by g' the 1D slice obtained by cutting the f_{i1D} at (x, y) along the orientation θ. Then, the phase of $f_{i1D}(x, y)$ is defined as the phase of the 1D signal g' evaluated at (x, y) using the analytic signal technique. With the aid of the monogenic signal, as illustrated in Figure 3.11 using a spherical coordinate system, θ and ϕ can be estimated in an isotropic way as [114]

$$\theta = \arctan\frac{h_y\{f_{i1D}\}}{h_x\{f_{i1D}\}}, \theta \in [0, \pi) \tag{3-27}$$

$$\phi = \arctan 2\left(\operatorname{sgn}(h_x\{f_{i1D}\})\sqrt{(h_x\{f_{i1D}\})^2 + (h_y\{f_{i1D}\})^2}, f_{i1D}\right), \phi \in [0, 2\pi) \tag{3-28}$$

where $\operatorname{sgn}(x)$ returns the sign of x.

To summarize, we can draw an analogy between the analytic signal and the monogenic signal, and such a comparison is listed in Table 3.2, which is adapted from [131].

C. Higher Order Riesz Transforms

As stated, in case of 2D image signals, the 1st-order Riesz transform based monogenic signal enables the rotationally invariant analysis of i1D structures, such as edges and lines. In order to characterize i2D image structures, e.g., corners and junctions, higher order Riesz transforms need to be exploited [120, 123, 124]. In this paper, we only consider 2nd-order Riesz transforms. Given an image $f(\mathbf{x})$, the three 2nd-order Riesz transforms of f are defined as

$$h_{xx}\{f\}(\mathbf{x}) \equiv h_x\{h_x\{f\}\}(\mathbf{x}) \tag{3-29}$$

$$h_{xy}\{f\}(\mathbf{x}) \equiv h_x\{h_y\{f\}\}(\mathbf{x}) \tag{3-30}$$

$$h_{yy}\{f\}(\mathbf{x}) \equiv h_y\{h_y\{f\}\}(\mathbf{x}) \tag{3-31}$$

Using the convolution theorem, the transfer functions of h_{xx}, h_{xy}, and h_{yy} in the Fourier domain are

$$H_{uu}(\mathbf{u}) \equiv \left(-j\frac{u}{\sqrt{u^2+v^2}}\right)\left(-j\frac{u}{\sqrt{u^2+v^2}}\right) = -\frac{u^2}{u^2+v^2} \tag{3-32}$$

$$H_{uv}(\mathbf{u}) \equiv \left(-j\frac{u}{\sqrt{u^2+v^2}}\right)\left(-j\frac{v}{\sqrt{u^2+v^2}}\right) = -\frac{uv}{u^2+v^2} \tag{3-33}$$

$$H_{vv}(\mathbf{u}) \equiv \left(-j\frac{v}{\sqrt{u^2+v^2}}\right)\left(-j\frac{v}{\sqrt{u^2+v^2}}\right) = -\frac{v^2}{u^2+v^2} \tag{3-34}$$

3.3.2 Encoding Local Patterns Using Riesz Transforms

In this sub-section, we will present two novel coding methods, namely RCode1 and RCode2, based on the image's responses to the Riesz transforms.

A. Computing Single-Band Riesz Transforms of a Given Image

Riesz transforms based image analysis, e.g., the monogenic signal, assumes that the signal consists of few frequencies, or in other words, it is band limited. However, real images usually consist of a wide range of frequencies. Therefore, it is necessary to pre-filter the image with a chosen band-pass filter before applying the Riesz transform to it. Based on the associative property of the convolution operation, we can apply the Riesz transforms to the chosen band-pass filter and then to filter the image with these new filters. With respect to the band-pass filter, there are many candidates proposed in the literature and we choose the widely used Gabor filter h_{BP} [38, 107], in this thesis. The transfer function of h_{BP} in the Fourier domain is defined as

$$H_{BP}(\mathbf{u}) = \begin{cases} 0, \mathbf{u} = (0,0) \\ \exp\left(-\frac{(|\mathbf{u}| - \mu_0)^2}{2\zeta^2}\right), else \end{cases} \tag{3-35}$$

where μ_0 and ς are two parameters to control the shape of the Gabor filter in the frequency domain.

We can apply the filters $h_x\{h_{BP}\}$, $h_y\{h_{BP}\}$, $h_{xx}\{h_{BP}\}$, $h_{xy}\{h_{BP}\}$ and $h_{xy}\{h_{BP}\}$ to the image to get its responses to the various Riesz transforms after a band-pass filtering. In the Fourier domain, the transfer functions of $h_x\{h_{BP}\}$, $h_y\{h_{BP}\}$, $h_{xx}\{h_{BP}\}$, $h_{xy}\{h_{BP}\}$ and $h_{yy}\{h_{BP}\}$ can be expressed as $H_{BP}{\cdot}H_u$, $H_{BP}{\cdot}H_v$, $H_{BP}{\cdot}H_{uu}$, $H_{BP}{\cdot}H_{uv}$, $H_{BP}{\cdot}H_{vv}$. Shapes of these filters (with a specific μ_0 and ς) in the spatial domain are shown in Figure 3.6. Moreover, according to the formula Eq. (3-20), the expression for the monogenic signal of a band-pass filtered image will be

$$f_{Mb}(\mathbf{x}) = \left(h_{BP}\{f\}(\mathbf{x}), h_x\{h_{BP}\{f\}\}(\mathbf{x}), h_y\{h_{BP}\{f\}\}(\mathbf{x})\right) \tag{3-36}$$

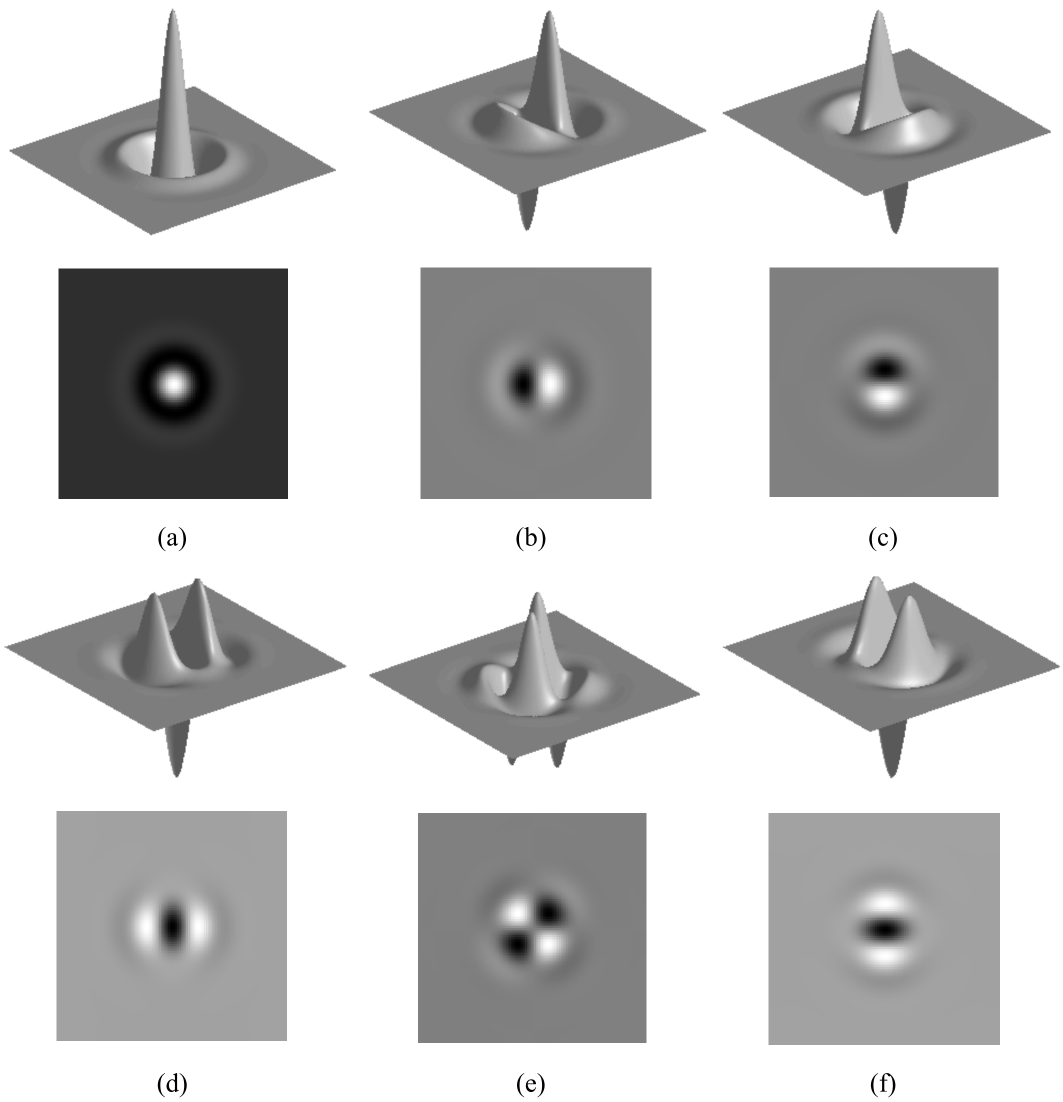


Figure 3.12 Shapes of the filters in the spatial domain. They are used to calculate the Riesz transforms of an image. For each filter, its shape is shown both in 3D surface format and in image format. (a) h_{BP}; (b) $h_x\{h_{BP}\}$; (c) $h_y\{h_{BP}\}$; (d) $h_{xx}\{h_{BP}\}$; (e) $h_{xy}\{h_{BP}\}$; (f) $h_{yy}\{h_{BP}\}$.

B. RCode1

The first Riesz-transforms-based coding method proposed, RCode1, is based on the image's responses to the filters h_{BP}, $h_x\{h_{BP}\}$, and $h_y\{h_{BP}\}$. Specifically, given an image $f(\mathbf{x})$, its corresponding RCode1($\mathbf{x}$) is three bits obtained by binarizing $h_{BP}\{f\}(\mathbf{x})$, $h_x\{h_{BP}\{f\}\}(\mathbf{x})$, and $h_y\{h_{BP}\{f\}\}(\mathbf{x})$ according to their signs. From Eq. (3-36), it can be seen that $h_{BP}\{f\}(\mathbf{x})$, $h_x\{h_{BP}\{f\}\}(\mathbf{x})$, and $h_y\{h_{BP}\{f\}\}(\mathbf{x})$ are the three components of the monogenic signal $f_{Mb}(\mathbf{x})$. Thus, RCode1($\mathbf{x}$) can represent the octant of $f_{Mb}(\mathbf{x})$ in the spherical coordinate system as illustrated in Figure 3.11. Hence, RCode1($\mathbf{x}$) can roughly reflect the local orientation and the local phase of $f(\mathbf{x})$ if $f(\mathbf{x})$ is regarded locally as an i1D signal. The computation process of RCode1 is illustrated in Figure 3.13 using an FKP image taken from [109]. Example RCode1 maps extracted from FKP images are shown in Figure 3.14.

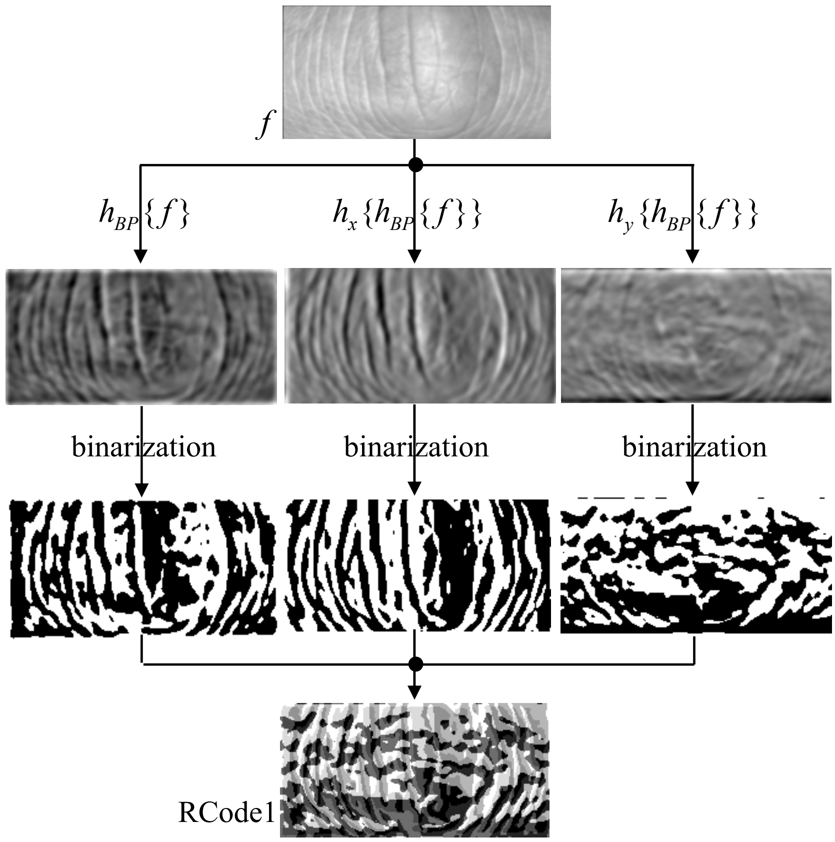


Figure 3.13 Illustration for the RCode1 computation process for a given finger-knuckle-print image.

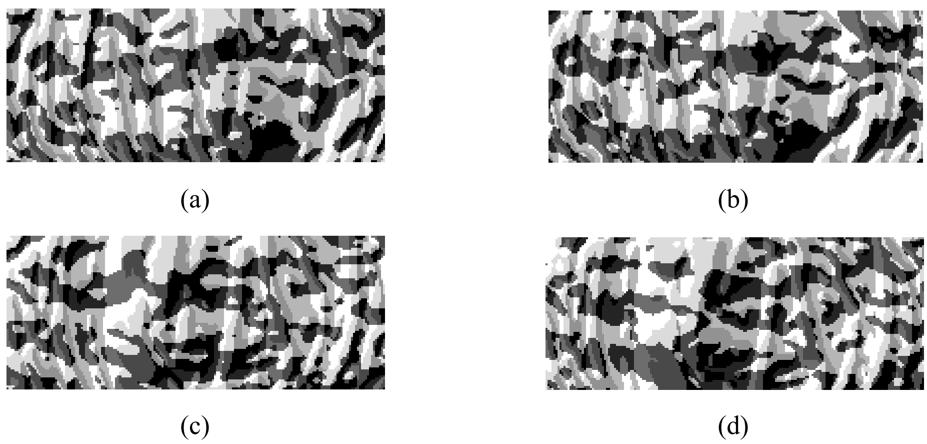


Figure 3.14 (a) ~ (d) RCode1 maps for the FKP ROI images shown in Figure 2.6, respectively.

Consider two RCode1 maps, P and Q. $P_1(Q_1)$, $P_2(Q_2)$, and $P_3(Q_3)$ are three bit-planes of P (Q). Then, we adopt the normalized Hamming distance to measure the dissimilarity between P and Q as

$$d(P,Q)=\frac{\sum_{y=1}^{Rows}\sum_{x=1}^{Cols}\sum_{i=1}^{3}P_i(x,y)\otimes Q_i(x,y)}{3S} \tag{3-37}$$

where S is the area of the code map, and $\otimes$ represents the bitwise "exclusive OR" operation.

B. RCode2

Similar as deriving RCode1, by binarizing $h_{xx}\{h_{BP}\{f\}\}$, $h_{xy}\{h_{BP}\{f\}\}$ and $h_{yy}\{h_{BP}\{f\}\}$ according to their signs, we can get another 3-bits coding scheme RCode2 for a given image f. The computation process of RCode2 is illustrated in Figure 3.15 using an FKP image. Example RCode2 maps extracted from FKP images are shown in Figure 3.16. Given two RCode2 maps, their dissimilarity can also be computed according to the formula Eq. (3-37).

In fact, since each bit of RCode1 and RCode2 is obtained by binarizing the image's responses to a filter, it can be regarded as an ordinal measure [30] of the local image structure. Thus, RCode1 and RCode2 should have the common advantages of ordinal measures, e.g., robust to illumination changes [30].

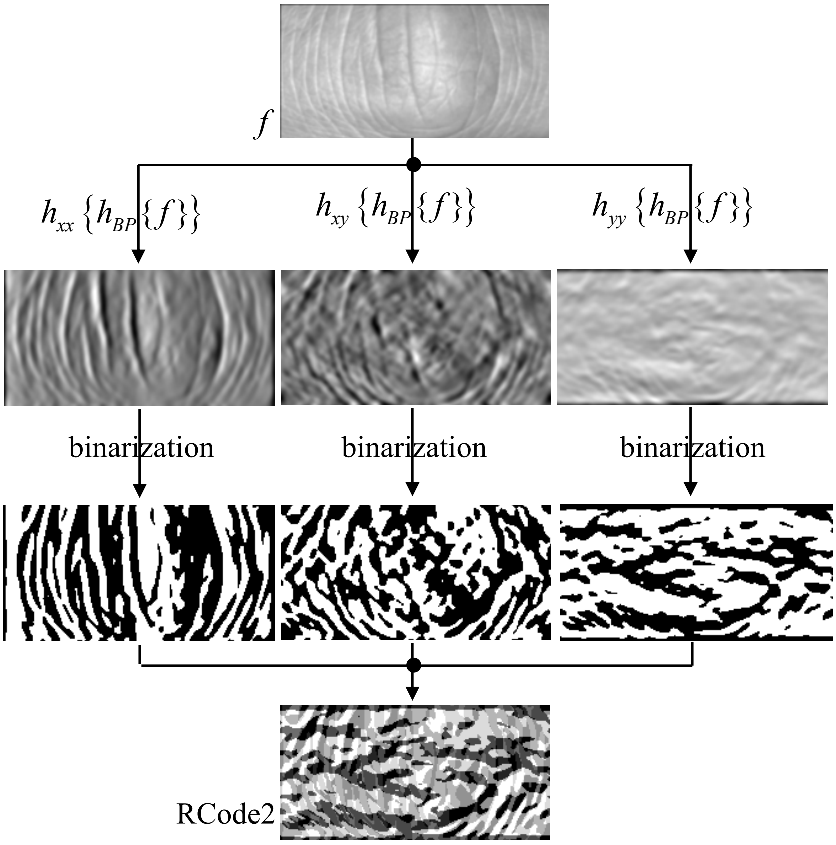
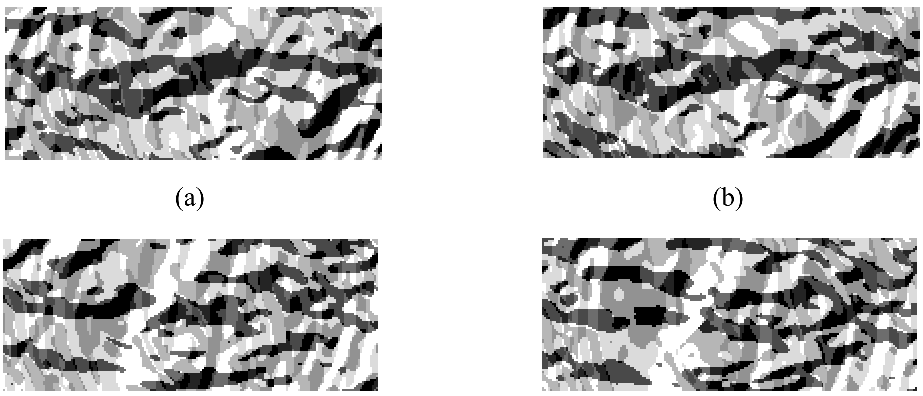


Figure 3.15 Illustration for the RCode2 computation process for a given finger-knuckle-print image.



(a) (b)

(c) (d)

Figure 3.16 (a) ~ (d) RCode2 maps for the FKP ROI images shown in Figure 2.6, respectively.

3.4 Experimental Results

3.4.1 Test Protocol

Experiments were conducted on the collected benchmark FKP database [109]. In that database, sample images for each subject were collected in two sessions. In our experiments, we took images collected in the first session as the gallery set and images collected at the second session as the probe set. Therefore, there were 660 (165 × 4) classes and 3,960 (660 × 6) images in the gallery set and the probe set each. To obtain statistical results, each image in the probe set was matched with all the images in the gallery set. If the two images were from the same class, the matching between them was counted as a genuine matching; otherwise it was counted as an imposter matching. Thus, the numbers of genuine matchings and imposter matchings were 23,760 and 15,657,840, respectively.

The equal error rate (EER), which is the point where the false accept rate (FAR) is equal to the false reject rate (FRR), is used to evaluate the verification accuracy. Besides, the receiver operating characteristic (ROC) curve, which is a plot of FRR against FAR for all possible thresholds, obtained by using each evaluated coding method will be provided.

3.4.2 FKP Verification Results

Table 3.3 Evaluation of six coding methods for FKP verification.

Coding Method	Parameters Settings	EER (%)
CompCode	$\sigma_x = 5.0$, $\sigma_y = 9.0$	1.658
RLOC	$lw = 4$	1.912
OrdinalCode	$s_x = 1.2$, $s_y = 4.2$	3.405
ImCompCode&MagCode	$\sigma_x = 5.0$, $\sigma_y = 9.0$, $N = 8$, $T = 0.2$, $w = 31$	1.475
RCode1	$\mu_0 = 0.085$, $\varsigma = 0.033$	1.661
RCode2	$\mu_0 = 0.084$, $\varsigma = 0.030$	1.610

All the coding based feature extraction and matching methods, CompCode, RLOC, OrdinalCode, ImCompCode&MagCode, RCode1, and RCode2, were evaluated in experiments. It should be noted that there are several parameters need to be tuned for each method. To this end, a sub-dataset, which contained the first 200 FKP classes, was used to adjust the parameters required for each evaluated method. Parameter settings and verification accuracies for all the evaluated methods are summarized in Table 3.3. Figure 3.17 shows the ROC curves generated by the six coding schemes.

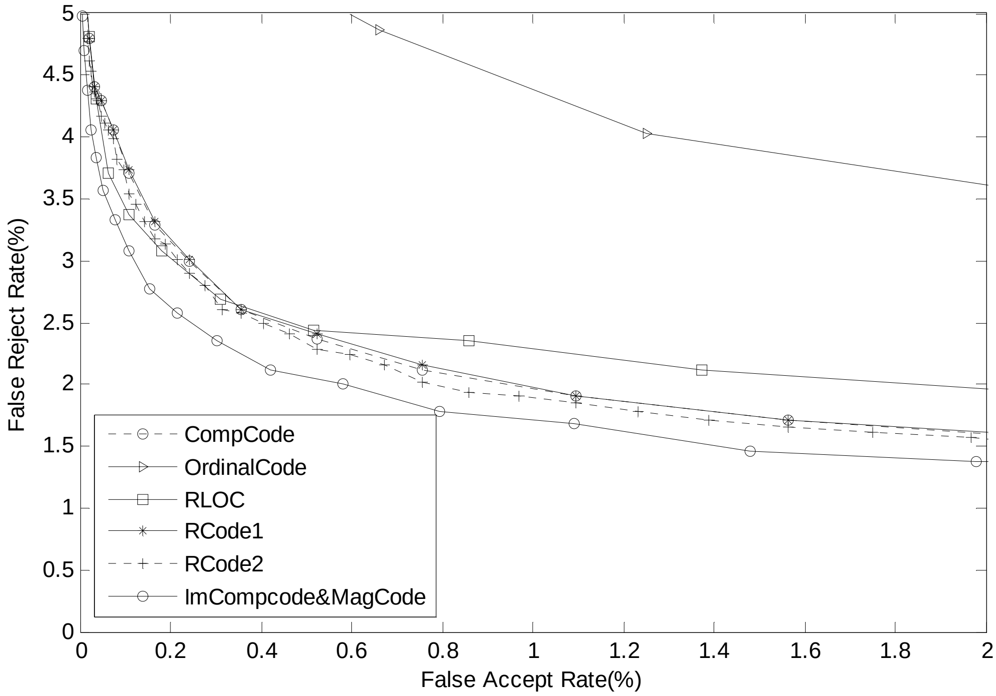


Figure 3.17 ROC curves obtained by the six coding methods on PolyU finger-knuckle-print database.

3.4.3 Discussion

From the verification results listed in Table 3.3 and the ROC curves shown in Figure 3.17, we can have the following findings: 1) the proposed method ImCompCode&MagCode could achieve better performance than all the other methods evaluated for FKP verification; 2) the proposed Riesz transforms based coding methods, RCode1 and RCode2, could achieve comparable verification performance with the other state-of-the-art coding methods.

In addition to the verification accuracy, we'd like to evaluate the mentioned various coding methods from several other aspects, including the feature size, the feature extraction speed, and the feature matching speed. The characteristics of these coding methods are summarized in Table 3.4. Each method was implemented with Visual C#.Net 2005 on a Dell Inspiron 530s PC embedded Intel 6550 processor and 2GB RAM. Convolutions were accomplished via the FFT (Fast Fourier Transform) according to the convolution theorem.

Table 3.4 Characteristics of each coding method.

	Code Bits	Extraction Time (msec) for 1 FKP Image	Time for 1 matching (msec)
CompCode	3	54.6	1.2
RLOC	3	54.6	1.2
OrdinalCode	3	17.6	1.2
ImCompCode&MagCode	6	105	1.6
RCode1	3	17.6	1.2
RCode2	3	17.6	1.2

From Table 3.4 we can see that ImCompCode&MagCode needs more bits to represent each code than the other coding methods and it needs more time for the feature extraction and matching. Except ImCompCode&MagCode, all the other five coding methods evaluated use 3-bits to represent each code, and since they all make use of the normalized Hamming distance at the matching stage, they have the same computation cost for one matching.

Of all the 3-bits coding methods, CompCode and RCode2 have similar performance and are better than the others in terms of the verification accuracy, which can be reflected from Table 3.3. However, these two methods have different computation complexity at the feature extraction stage. Since the major operations at the feature extraction stage are the convolutions, the number of convolutions can roughly reflect the overall feature extraction complexity. RCode2 needs three convolutions while CompCode needs six. In fact, for CompCode, an extra operation is needed to figure out the minimum of the filters' responses at each position. From Table 3.4, we can see that RCode2 is nearly three times faster than CompCode at the feature extraction stage. Thus, we claim that the proposed RCode2 method is the best 3-bits coding method and can serve as a better candidate for real-time applications.

3.5 Summary

This chapter focuses on coding-based algorithms for the feature extraction and matching of FKP images. Based on the framework of CompCode, we present a new coding method, namely ImCompCode&MagCode, which considers not only the orientation information but also the magnitude information extracted by the Gabor filters. Furthermore, based on the Riesz transforms, we present two 3-bits coding schemes, RCode1 and RCode2. The proposed coding methods are evaluated on the collected FKP database along with several other state-of-the-art coding schemes. Based on the experimental results, we conclude that ImCompCode&MagCode performs the best in terms of the verification accuracy while RCode2 is the best 3-bits coding method for FKP recognition.

Chapter 4. Feature Extraction and Matching II: A Fourier Phase-based Method

In the last chapter, we have discussed coding-based feature extraction and matching methods for FKP recognition. In fact, coding-based methods make use of local information contained in images. Actually, global information hidden in images could also be explored for the recognition purpose. In this chapter, we present a global-based FKP recognition scheme. In this method, we use Fourier transform as a global feature extractor and use the band-limited phase-only correlation technique to compute the similarity between the Fourier coefficients of two images.

4.1 Phase-only Correlation (POC)

In this method, we take the Fourier transform coefficients of images as features and resort to the phase-only correlation (POC) technique [106] to measure their similarities. In the literature, POC based methods have been widely used in image registration tasks. The importance of the phase of the Fourier transform in image reconstruction was systematically investigated in the salient work done by Oppenheim and Lim [136]. Many efforts have been devoted in making use of phase information for image matching and image registration, especially in optical image processing systems where the analogue Fourier transforms could be obtained efficiently. Early Fourier phase based image matching or registration methods could only deal with the translation between images [106]. Later developed models could deal with rotation, scaling, or even affine transformations between images [137-144]. According to [144], generally speaking, Fourier-phase based image matching or registration methods can have the following advantages over the other algorithms:

- Fourier-based schemes are very robust to noise and time varying illumination changes.

- They have a low computational complexity and take a fixed period of time for registering or matching any images. By contrast, most spatial based registration methods usually have high computational complexities and the time used in registering two images depends on the content of the images to be registered.
- Fourier-phase based methods are easy to be efficiently implemented due to the existence of the FFT algorithm.

Recently, POC has also been adopted as a similarity measure to match some kinds of biometric images, such as iris images [29], fingerprint images [42-44], and palmprint images [60]. Compared with the conventional POC, the Band-Limited Phase-Only Correlation (BLPOC) proposed by Ito *et al*. [42] is more effective. Hence, in this thesis, we use BLPOC to evaluate the displacement parameters between FKP ROIs and to measure the similarity of the Fourier transforms of the aligned ROIs. In this sub-section, POC will be introduced and in the next sub-section BLPOC will be described.

POC is a kind of effective method to evaluate the translation parameters between two images in the Fourier domain. Its underlying principle is the translation property of the Fourier transforms [145]. Let f and g be the two images that differ only by a displacement (x_0, y_0), i.e.

$$g(x,y) = f(x - x_0, y - y_0) \tag{4-1}$$

Their corresponding Fourier transforms $G(u,v)$ and $F(u,v)$ will be related by

$$G(u,v) = e^{-j2\pi(ux_0 + vy_0)} F(u,v) \tag{4-2}$$

The cross-phase spectrum $R_{GF}(u,v)$ between $G(u,v)$ and $F(u,v)$ is given by

$$R_{GF}(u,v) = \frac{G(u,v)F^*(u,v)}{\left|G(u,v)F^*(u,v)\right|} = e^{-j2\pi(ux_0 + vy_0)} \tag{4-3}$$

where F^* is the complex conjugate of F. By taking inverse Fourier transform of R_{GF} back to the spatial domain, we will have a Dirac impulse centered on (x_0, y_0).

In practice, we should consider the finite discrete representations. Consider two $M \times N$

images, $f(m, n)$ and $g(m, n)$, where the index ranges are $m = -M_0, \ldots, M_0$ ($M_0 > 0$) and $n = -N_0, \ldots, N_0$ ($N_0 > 0$), and $M = 2M_0 + 1$ and $N = 2N_0 + 1$. Denote by $F(u, v)$ and $G(u, v)$ the 2D DFTs of the two images and they are given by

$$F(u,v) = \sum_{m=-M_0}^{M_0} \sum_{n=-N_0}^{N_0} f(m,n) e^{-j2\pi\left(\frac{mu}{M}+\frac{nv}{N}\right)} = A_F(u,v) e^{j\phi_F(u,v)} \quad (4\text{-}4)$$

$$G(u,v) = \sum_{m=-M_0}^{M_0} \sum_{n=-N_0}^{N_0} g(m,n) e^{-j2\pi\left(\frac{mu}{M}+\frac{nv}{N}\right)} = A_G(u,v) e^{j\phi_G(u,v)} \quad (4\text{-}5)$$

where $u = -M_0, \ldots, M_0$, $v = -N_0, \ldots, N_0$, $A_F(u, v)$ and $A_G(u, v)$ are amplitude components, and$\varphi_F(u, v)$ and$\varphi_G(u, v)$ are phase components. Then, the cross-phase spectrum $R_{GF}(u, v)$ between $G(u, v)$ and $F(u, v)$ is given by

$$R_{GF}(u,v) = \frac{G(u,v)F^*(u,v)}{\left|G(u,v)F^*(u,v)\right|} = e^{j\{\phi_G(u,v)-\phi_F(u,v)\}} \quad (4\text{-}6)$$

The POC function $p_{gf}(m, n)$ is the 2D Inverse DFT (IDFT) of $R_{GF}(u, v)$:

$$p_{gf}(m,n) = \frac{1}{MN} \sum_{u=-M_0}^{M_0} \sum_{v=-N_0}^{N_0} R_{GF}(u,v) e^{j2\pi\left(\frac{mu}{M}+\frac{nv}{N}\right)} \quad (4\text{-}7)$$

The peak value of p_{gf} can be calculated as $\max\{p_{gf}(m, n) :| m \in [-M_0, M_0], n \in [-N_0, N_0]\}$

If the two images f and g are similar, their POC function p_{gf} will give a distinct sharp peak. If not, the peak value will drop significantly. Thus, the amplitude of the peak value can be used as a similarity measure, and the location of the peak shows the translational displacement between the two images.

4.2 Band-limited Phase-only Correlation (BLPOC)

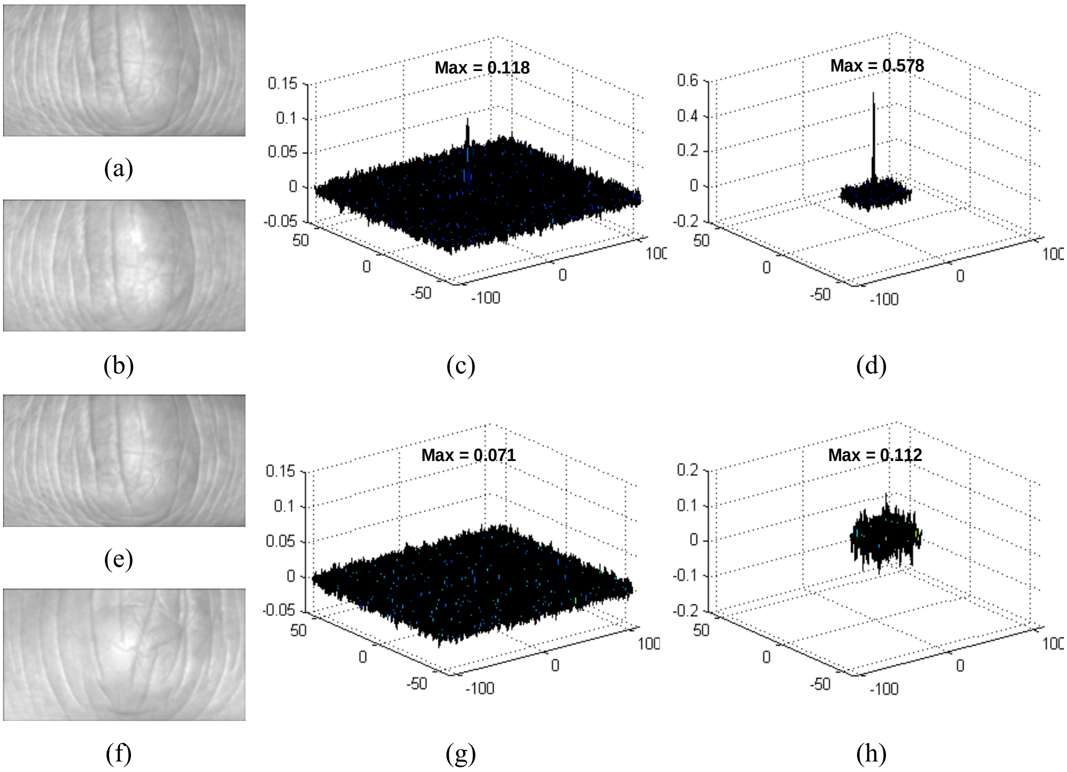


Figure 4.1 Examples of a genuine matching and an imposter matching using POC and BLPOC respectively. (a) and (b) are two FKP ROI images from the same finger, (c) is their POC function and (d) is their BLPOC function. (e) and (f) are two FKP ROI images from different fingers, (g) is their POC function and (h) is their BLPOC function.

In the POC-based image matching method, all the frequency components are involved. However, high frequency components can be prone to noise. To eliminate noisy high frequency components, Ito *et al.* [42] proposed the Band-Limited POC (BLPOC). BLPOC limits the range of the spectrum of the given FKP image. Suppose that the ranges of the inherent frequency band of FKP texture are given by $u = -U_0, \ldots, U_0$ and $v = -V_0, \ldots, V_0$, where $0 \le U_0 \le M_0$, $0 \le V_0 \le N_0$. Thus, the effective size of spectrum is given by $L_1 = 2U_0 + 1$ and $L_2 = 2V_0 + 1$. BLPOC function is defined as

$$p_{gf}^{U_0V_0}(m,n) = \frac{1}{L_1L_2}\sum_{u=-U_0}^{U_0}\sum_{v=-V_0}^{V_0} R_{GF}(u,v)e^{j2\pi\left(\frac{mu}{L_1}+\frac{nv}{L_2}\right)} \tag{4-8}$$

where $m = -U_0, \ldots, U_0$ and $n = -V_0, \ldots, V_0$. From the definition of BLPOC, we can see that

U_0/M_0 and V_0/N_0 can characterize the inherent frequency distribution of the FKP images.

From the definition of BLPOC, it can be seen that the BLPOC function between two images f and g can be considered as the POC function between their low-pass filtered versions. Thus, the BLPOC function can maintain the properties of the POC function. Specifically, if two images are similar, their BLPOC function will have a distinct sharp peak. At the same time, the translational displacement between the two images can be estimated by the location of the peak. Experiments indicate that the BLPOC function provides a much higher discrimination capability than the original POC function in FKP recognition. This can be reflected in the matching examples shown in Figure 4.1. Figure 4.1(a) and Figure 4.1(b) are two FKP ROI images from the same finger (captured in different collection sessions), whose POC function and BLPOC function are shown in Figure 4.1(c) and Figure 4.1(d), respectively; Figure 4.1(e) and Figure 4.1(f) are two FKP ROI images from different fingers, whose POC function and BLPOC function are shown in Figure 4.1(g) and Figure 4.1(h), respectively. These examples indicate that in the case of a genuine matching (a matching performed between a pair of FKP images from the same finger), the BLPOC will exhibit a much sharper peak than POC; however, for an imposter matching (a matching performed between a pair of FKP images from different fingers), neither BLPOC nor POC will show a distinct sharp peak. Hence, in this thesis, we adopt the BLPOC to align the displacement between FKP ROI images and then to measure the similarity between Fourier transforms of the aligned ROIs.

4.3 BLPOC based FKP Recognition Algorithm

In this sub-section, we will describe details of our FKP recognition algorithm based on the BLPOC technique introduced in section 4.2. The proposed algorithm consists of 3 basic steps: translation alignment, common regions extraction, and matching score calculation. The whole process is summarized in Table 4.1 and is also illustrated in Figure 4.2.

Table 4.1 FKP recognition using BLPOC

Input: two FKP ROI images f and g
Output: the matching score between f and g
1. estimate the translational displacement (t_1, t_2) between f and g using BLPOC;
2. align f and g using (t_1, t_2) and then extract the common-regions, f_C and g_C;
3. calculate the BLPOC function between f_C and g_C, denoted by P_{fg};
4. take the highest peak of P_{fg} as the matching score.
Begin module
End module

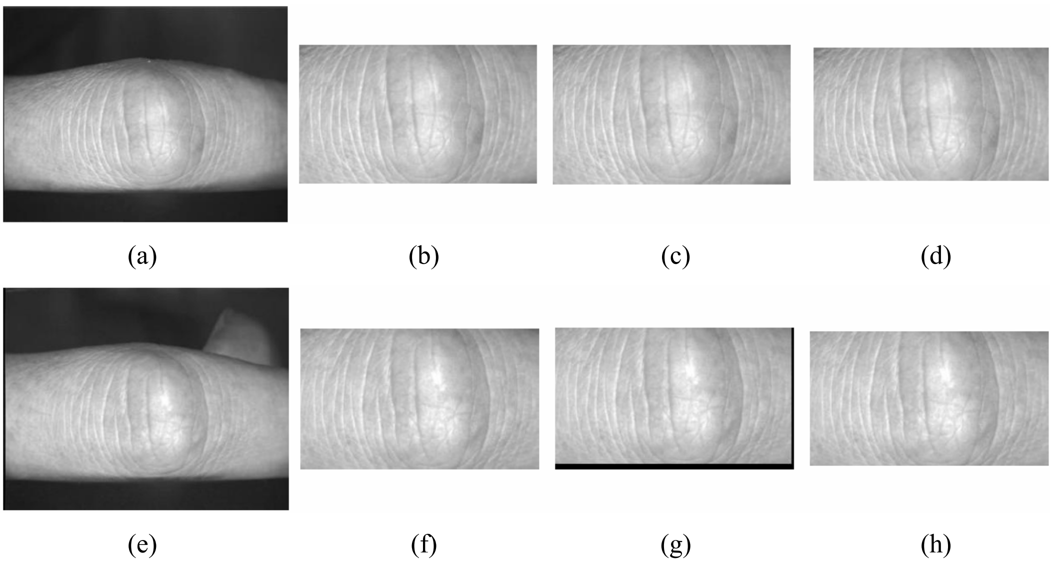


Figure 4.2 Intermediate steps of our FKP recognition algorithm: (a) and (e) are two FKP images; (b) and (f) are ROIs image f and g, extracted from (a) and (e), respectively; (c) and (g) are the registered versions of f and g using the translation parameters estimated by the BLPOC function between f and g; (d) and (h) are the common regions f_C and g_C, extracted from registered f and g as shown in (c) and (g). Then, we can calculate the matching score using the BLPOC function between f_C and g_C.

4.3.1 Translation Alignment and Common Regions Extraction

The purpose of this step is to enhance genuine matching scores while not increasing much imposter matching scores to improve the overall recognition accuracy of the system. Consider two FKP ROI images, f and g, from the same finger but captured in different collection sessions. Although the design of the image acquisition device can reduce much the position variation of the user's finger, it is still inevitable that there is some displacement between f and g, and the non-overlapped areas of them would make their

BLPOC function noisy. Thus, in order to reduce the negative effects of the non-overlapped regions, we need to extract the common areas, denoted by f_C and g_C, from f and g.

At first, the displacement pair (t_1, t_2) between f and g can be estimated from the peak location of the BLPOC function of them. Then, we can align f and g based on the displacement pair (t_1, t_2) and extract the common regions f_C and g_C. Such a process is illustrated in Figure 4.2.

It should be noted that in our system, we will check the ratio between the common region area and the area of the effective region. If area(f_C)/area(f) < t (or area(g_C)/area(g) < t), where t is a threshold, f_C and g_C. will be directly set to f and g. Generally, this will happen when the two FKP images are from different fingers.

4.3.2 Matching Score Calculation

Having obtained f_C and g_C, it is time for us to calculate the matching score now. The matching score is taken as the highest peak of the BLPOC function of f_C and g_C. In verification, if the matching score between FKP images is higher than a preset threshold, the two FKP images are considered to be from the same finger.

4.4 Experiments

In order to test the efficacy of the proposed BLPOC-based FKP recognition algorithm, we performed experiments on the collected benchmark FKP database.

4.4.1 Determination of Parameters

Table 4.2 Parameter settings for the BLPOC based FKP recognition method.

U_0/M_0	V_0/N_0	t
0.25	0.2	0.75

It should be noted that there are several parameters need to be tuned for this method. To this end, a sub-dataset, which contained the first 200 FKP classes, was used to adjust the parameters required. Parameter settings are summarized in Table 4.2

4.4.2 FKP Verification Results

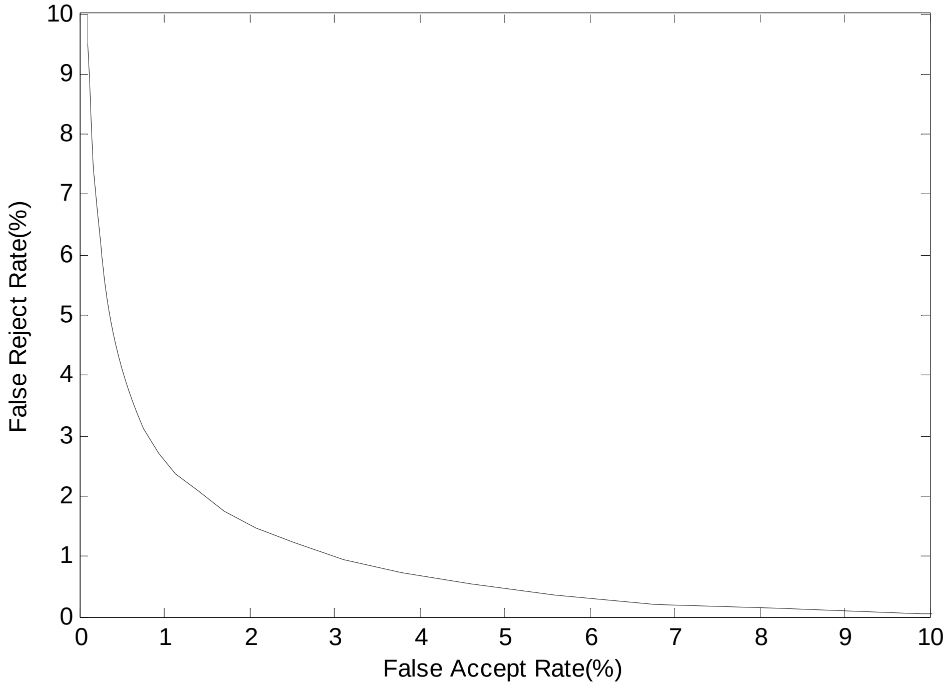


Figure 4.3 ROC curve obtained by the BLPOC FKP verification method.

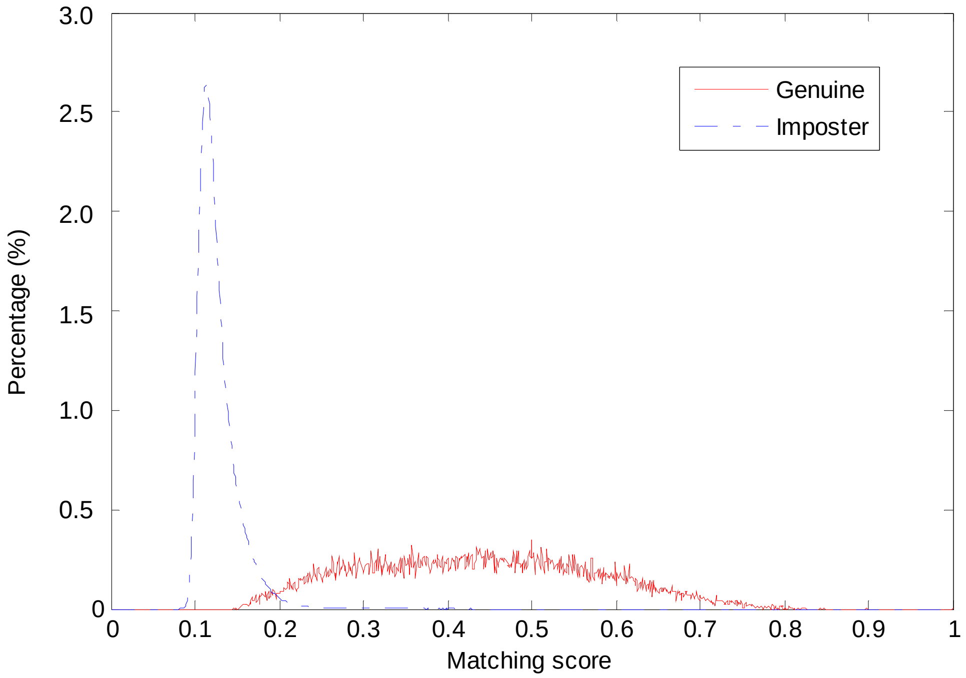


Figure 4.4 Distance distributions of genuine matchings and imposter matchings obtained by the BLPOC FKP verification method.

The test protocols and evaluation measures are the same as mentioned in section 3.4.1. The EER obtained by this method is 1.68%. Figure 4.3 shows the ROC curve and Figure 4.4 shows the distance distributions of genuine matchings and imposter matchings generated by this method.

4.5 Summary

In this chapter, we present a novel global-feature based FKP recognition method. Specifically, Fourier transform coefficients of images are taken as features and the band-limited phase-only correlation technique is utilized to compute their similarity. Its efficacy is corroborated in the experiments. Performance comparison between this method and the other FKP recognition methods will be presented in the next chapter.

Chapter 5. Feature Extraction and Matching III: A Local-global-based Method

Based on the area of pixels involved in feature extraction, we can label the features as "local" or "global" ones. Intuitively, a local feature is a measure computed within a local patch, encoding the detailed traits within this specific area; by contrast, a global feature is a measure derived from all (or most of) the pixels in the image, reflecting some holistic characteristic of the examined image. According to such definitions, coding-based FKP recognition schemes discussed in Chapter 3 can be classified into local-based methods and the BLPOC-based approach presented in Chapter 4 is actually a global-based approach.

In the literature of psychophysics and neurophysiology, many studies have shown that both local and global information is crucial for the image perception and recognition of human beings [21] and they play different but complementary roles. A global feature reflects the holistic characteristics of the image and is suitable for coarse representation, while a local feature encodes more detailed information within a specific local region and is appropriate for finer representation. Hence, better recognition accuracy can be expected if local and global information can be appropriately combined.

Such an idea has already been explored in iris recognition, palmprint recognition, face recognition and fingerprint recognition. For iris matching, Sun *et al*. [26, 27] proposed a "cascade" system in which the first stage is a conventional Daugman-like classifier while the classifier at the second stage uses "global" features – areas enclosed by zero-crossing boundaries. In [55], the authors described a two-level palmprint matching scheme. For coarse-level filtering, Hough transform is used to extract global features; for fine-level matching, the local information extracted from the locations and orientations of individual lines is used. Pan *et al*. also proposed to combine the local and global features for palmprint recognition [56]. In their work, non-negative matrix factorization with sparseness constraint and PCA are used to extract local and global features, respectively. For face recognition, Fang *et al*. [13] presented a method by combining global PCA features and component-based local features extracted by Haar wavelets. In [21], Su *et al*. proposed a hierarchical ensemble classifier by combining global Fourier features and local Gabor features. In their method, global features are extracted from the whole face images by keeping the low-frequency Fourier coefficients while local features are exploited using Gabor filters with various scales and orientations. After that, Fisher's linear discriminant (FLD) is applied to the global Fourier features and local Gabor features. In the fingerprint recognition community, the idea of combining local and global information was also exploited [41].

In this chapter, with the belief that the recognition accuracy could be much improved by the fusion of local-global features, we propose a novel unified local-global feature extraction and fusion scheme for the verification of FKP images. The features used in our scheme are some low level image features defined under the time-frequency analysis framework. Specifically, the local features defined include local orientation, local phase and local phase congruency. They reflect different aspects of information within a local patch and none of them can be covered by the others. It will be seen that these three kinds of local features depend on the local time-frequency analysis of the image. In practice, quadrature pair of filters, such as complex Gabor [107] or log-Gabor [108] filters, can be used to do the local time-frequency analysis.

Suppose that we use the complex Gabor transform to extract the local features. By increasing the scale of the complex Gabor filters, more and more global information will be involved, yet the characterization of image local structures will be rapidly weakened. Particularly, if the scale of the Gabor transform is increased to infinity, the Gabor transform can be reduced to the Fourier transform of the whole image. In this case, no image local information can be extracted but we can get the finest resolution for image global frequency analysis. Thus, the Fourier transform coefficients are naturally taken as global features in this paper. With the global Fourier features, the alignment between intra-class palmprint (or FKP) ROIs can also be refined. At the matching stage, four matching distances can be computed by comparing the local features and the global features separately. Finally, the four matching distances are fused according to some fusion rule to get the final matching distance. To our knowledge, this is the first time to define and assemble such kinds of local and global features together for matching biometric images. The promising experimental results corroborated the efficacy of the proposed local-global feature extraction method.

5.1 Local Features

A point **x** on an image can be characterized by its "local features", which are derived from a local patch surrounding it. Before we define local features we need to have a model for the signal we are going to analyze. In our case, we are dealing with 2D FKP images. However, based on the observation, FKP images are actually a special kind of 2D images, in that they are abundant of line-like features. And these line-like features play dominant roles in distinguishing different individuals. Thus, in this chapter, we assume that the FKP images are locally i1D (intrinsic one dimensional) signals. For the concept of intrinsic dimension, please refer to the section 3.3.1.

Let's consider the one dimensional (1D) signal first. The analytic signal is an effective tool for 1D signal processing. Given a real 1D signal $f(x)$, the corresponding analytic signal is defined as

$$f_A(x) = f(x) + if_H(x) = f(x) + i\big(f(x) * h(x)\big) \tag{5-1}$$

where $h(x) = 1/\pi x$ refers to the Hilbert transform kernel in the spatial domain. The local amplitude and the local phase of the 1D analytic signal are defined as

$$a(x) = \sqrt{f^2(x) + f_H^2(x)}, \phi(x) = \arg f_A(x) \tag{5-2}$$

The local amplitude indicates the energetic information of the signal, while the local phase can be used to distinguish different local structures and it is independent of the local amplitude [146]. When the 1D signal is embedded into the 2D space, its orientation should be considered. Thus, the local amplitude, the local phase, and the local orientation are three independent measures to characterize a 2D image point.

As stated above, the local phase can reflect the type of local structures. However, we do not know whether such a structure is significant and stable. To address such an issue, we need to know whether the local phases over scales are consistent. We use the phase congruency [147-156], a dimensionless quantity, to measure the consistency of the local phases over scales. Based on the physiological and psychophysical evidence, it is found that visually discernable features coincide with those points having maximal phase congruency. Such a conclusion has been further corroborated by some recent studies in neurobiology using functional magnetic resonance imaging (fMRI) [156]. Phase congruency has an intriguing property that it is almost invariant to changes in image brightness or contrast.

Thus, within the local patch surrounding an image point **x**, four features – local amplitude, local phase, local orientation and local phase congruency, can be extracted and they reflect different aspects of information contained in this patch. However, we will not use local amplitude for recognition because it is not contrast invariant. Hence, the local phase, the local orientation and the local phase congruency will be chosen as three local features in this thesis.

In practice, for 2D images, these three local features can be extracted using complex Gabor filters, which are defined as Eq. (3-1) in section 3.1. In order to facilitate reading, we copy the Eq. (3-1) here as Eq. (5-3):

$$G(x,y)=\exp\left(-\frac{1}{2}\left(\frac{x'^2}{\sigma_x^2}+\frac{y'^2}{\sigma_y^2}\right)\right)\cdot\exp\left(i\frac{2\pi}{\lambda}x'\right) \tag{5-3}$$

where $x' = x\cos\theta + y\sin\theta$, $y' = -x\sin\theta + y\cos\theta$. In Eq. (5-3), λ represents the wavelength of the sinusoid factor, θ represents the orientation of the normal to the parallel stripes of the Gabor function, σ_x and σ_y are the standard deviations of the 2D Gaussian envelop. It can be seen from the definition that a Gabor filter is actually a Gaussian envelop modulated by a sinusoidal plane wave. The Gaussian envelop ensures that the convolution is dominated by the image patch near the center of the filter. Thereby, when an image is convolved with a Gabor filter, the information near the center of the Gaussian envelop is encoded, and by contrast, the information far away from the center of the Gaussian envelop will be neglected. Therefore, the Gabor filter is a local operator and can extract the information at a specific scale and a specific orientation within a local region.

To extract local orientation information, we make use of the competitive coding scheme. Denote by G_R (G_I) the real (imaginary) part of the Gabor filter G. With a series of G_Rs sharing the same parameters, except the parameter of orientation, the local orientation information of the image I at the position (x, y) can be extracted. Mathematically, local orientation is defined as

$$ori(x,y) = \arg\min_j \{I(x,y) * G_R(x,y,\theta_j)\} \tag{5-4}$$

where $\theta_j = j\pi/J$, $j = \{0,\ldots, J - 1\}$. In our work, we set J as 6.

The extraction of the local phase congruency using Gabor filters will be presented in section 5.1.1 in detail. Actually, phase congruency is a 1D concept. For 2D images, we can compute PC_{θ_j} along different orientations $\{\theta_j:| j = 0 \sim J\text{-}1\}$. Then we take the maximum of $\{PC_{\theta_j} :| j = 0 \sim J-1\}$ as the PC value at the examined position:

$$PC_2(x,y) = \max_j \{PC_{\theta_j}(x,y) :| j = 0 \sim J-1\} \tag{5-5}$$

We denote by θ_m the orientation along which the 1D phase congruency takes the maximum. Then, we apply Gabor filtering along θ_m and define the local phase as:

$$phase(x,y) = \arg\left(I(x,y) * G_R(x,y,\theta_m) + iI(x,y) * G_I(x,y,\theta_m)\right) \tag{5-6}$$

In practice, for the reason of the computational efficiency, we do not compute the three local features separately. Instead, we present a scheme based on the computation framework of phase congruency proposed in [151-153] to extract those features. So, in the following sub-sections, the phase congruency and its computation is described first.

5.1.1 Phase Congruency

Rather than assume a feature is a point of sharp changes in intensity, the phase congruency model postulates that features are perceived at points where the Fourier components are maximal in phase [147-156]. Phase congruency (PC) can be considered as a dimensionless measure for the significance of a structure independent of the signal energy. Since its advent, PC has been widely used in various applications, such as iris recognition [28], palmprint recognition [61], and face recognition [17]. The technique to calculate phase congruency used in this thesis is based on Kovesi's work [151-153].

We start from the 1D signal $f(x)$. Denote by M_n^e and M_n^o the even-symmetric and odd-symmetric filters at scale n and they form a quadrature pair. Responses of each quadrature pair to the signal will form a response vector at position x and scale n:

$$[e_n(x), o_n(x)] = [f(x) * M_n^e, f(x) * M_n^o] \tag{5-7}$$

The local amplitude at scale n is given by

$$A_n(x) = \sqrt{e_n^2(x) + o_n^2(x)} \tag{5-8}$$

and the local phase is given by

$$\phi_n(x) = atan2(o_n(x), e_n(x)) \tag{5-9}$$

These response vectors form the basis of our localized representation of the signal and the phase congruency can be derived from them.

Let $F(x) = \sum_n e_n(x)$ and $H(x) = \sum_n o_n(x)$. Then, the 1-D phase congruency can be computed as

$$PC(x) = \frac{E(x)}{\varepsilon + \sum_n A_n(x)} \tag{5-10}$$

where $E(x) = \sqrt{F^2(x) + H^2(x)}$ and ε is a small positive constant. We can also define the local phase as

$$Phase(x) = atan2\left(H(x), F(x)\right) \tag{5-11}$$

Actually, it is the average local phase over n scales.

For 2D images, we have to apply the 1D analysis over several orientations and combine the results in some way. In such case, 2D filters with the orientation selection property can be used, such as Gabor filters [107] or log-Gabor filters [108]. Let $\theta_j = j\pi / J$, $j = \{0,1,\ldots, J\text{-}1\}$ denote the orientation angle of the filter, where J is the number of orientations. By modulating n and θ_j and convolving with the 2D image, we can get a set of responses at each image point $\mathbf{x}$ as

$$\left[e_{n,\theta_j}(\mathbf{x}), o_{n,\theta_j}(\mathbf{x})\right] \tag{5-12}$$

The local amplitude of point $\mathbf{x}$ at scale n and along orientation θ_j is given by

$$A_{n,\theta_j}(\mathbf{x}) = \sqrt{e_{n,\theta_j}(\mathbf{x})^2 + o_{n,\theta_j}(\mathbf{x})^2} \tag{5-13}$$

The local energy along orientation θ_j is given by

$$E_{\theta_j}(\mathbf{x}) = \sqrt{F_{\theta_j}(\mathbf{x})^2 + H_{\theta_j}\left(\mathbf{x}\right)^2} \tag{5-14}$$

where $F_{\theta_j}(\mathbf{x}) = \sum_n e_{n,\theta_j}(\mathbf{x})$ and $H_{\theta_j}(\mathbf{x}) = \sum_n o_{n,\theta_j}(\mathbf{x})$. Then, the phase congruency along orientation θ_j is given by

$$PC_{\theta_j}(\mathbf{x}) = \frac{E_{\theta_j}(\mathbf{x})}{\varepsilon + \sum_n A_{n,\theta_j}(\mathbf{x})} \tag{5-15}$$

The average local phase along orientation θ_j is defined as

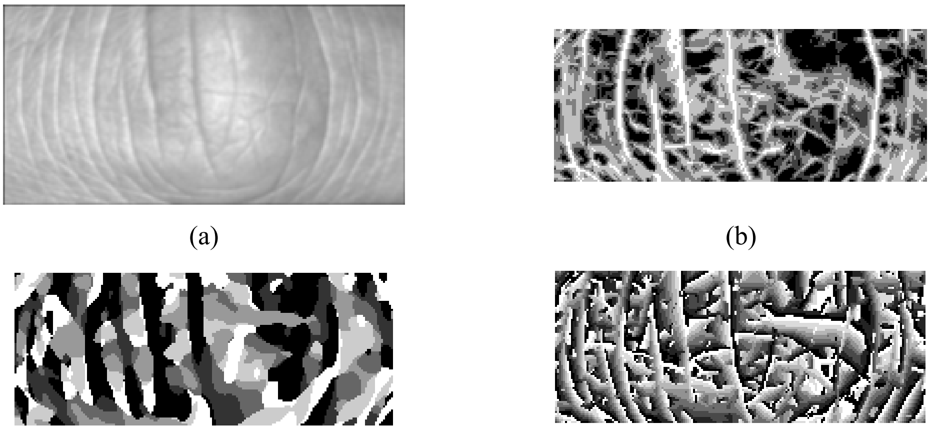
$$Phase_{\theta_j}(\mathbf{x}) = atan2\left(H_{\theta_j}(\mathbf{x}), F_{\theta_j}(\mathbf{x})\right) \tag{5-16}$$

We define the 2D phase congruency at $\mathbf{x}$ as

$$PC_2(\mathbf{x}) = \max_j PC_{\theta_j}(\mathbf{x}) \tag{5-17}$$

It should be noted that $PC_2(\mathbf{x})$ is a real number within $0 \sim 1$.

5.1.2 Local Feature Extraction and Coding



(a) (b)

(c) (d)

Figure 5.1 Examples of local feature maps. (a) is the original finger-knuckle-print ROI image; (b) is the corresponding *pcCode* map; (c) is the corresponding *oriCode* map; (d) is the corresponding *phaCode* map.

We have discussed the definition and computation of local phase congruency in the previous sub-section. Having obtained the two raw phase congruency maps of two images, we do not match them directly. Instead, we quantize them to several levels and then code them into integers. In practice, such a scheme can have three advantages: a) it can save a lot of storage; b) for recognition, it works more robustly than using raw phase congruency maps; and c) it allows a fast matching of two maps. Therefore, we quantize the phase congruency into L levels and define the phase congruency code as

$$pcCode(\mathbf{x}) = \left\lfloor \frac{PC_2(\mathbf{x})}{1/L} \right\rfloor \tag{5-18}$$

where $\lfloor x \rfloor$ is the operator to return the largest integer not bigger than x. It is easy to see that each *pcCode* is an integer within $0 \sim L$-1.

Although there are other kinds of methods to evaluate the local phase feature and the local orientation feature, we want to make a full use of the intermediate results in the process of computing phase congruency in order to reduce the computation cost. It is easy to see that when calculating the phase congruency, we can get responses from a set of even-symmetric and odd-symmetric quadrature filters at different scales and different orientations. We can compute the local orientation and the local phase directly from them. For local orientation evaluation, we borrow the idea from the competitive coding scheme [51]. With the responses from the even-symmetric filters at a certain scale ς, i.g. $e_{\zeta,\theta_j}(\mathbf{x})$, the orientation code at $\mathbf{x}$ can be defined as

$$oriCode(\mathbf{x}) = \arg\min_j \left\{ e_{\zeta,\theta_j}(\mathbf{x}) \right\}, j = 0, ..., J-1 \tag{5-19}$$

Obviously, each orientation code $oriCode(\mathbf{x})$ is an integer within 0 ~ J-1.

Refer to Eq. (5-17), by our definition the 2D phase congruency is actually the maximum of the 1D phase congruencies along different orientations. We denote by θ_m the orientation along which the 1D phase congruency takes that maximum. Then, we can take the average local phase along θ_m as the local phase at $\mathbf{x}$. That is

$$LP(\mathbf{x}) = Phase_{\theta_m}(\mathbf{x}) \tag{5-20}$$

The range of LP is [0, 2π]. Please note that the local phase defined in Eq. (5-20) is a bit different from the definition in Eq. (5-6), in that Eq. (5-6) defines phase using the Gabor filters at a single scale while Eq. (5-20) defines the local phase as the average phase over several scales. Usually, the average phase over several scales performs better than the phase defined on a single scale for the FKP recognition. So, in real implementation, we use Eq. (5-20) to calculate the local image phase. Once again, we do not need the exact local phase angle. Instead, we quantize LP into several discrete levels to get the "phase code" as

$$phaCode(\mathbf{x}) = \left\lfloor LP(\mathbf{x}) / \left(2\pi / M\right) \right\rfloor \tag{5-21}$$

where M is the number of quantization levels. Thus, each phase code is an integer within 0 ~ M-1.

Finally, for a given image, we can get its three code maps: *pcCode*, *oriCode*, and *phaCode*. Examples of local feature code maps extracted from an FKP image are shown in Figure 5.1.

5.1.3 Matching of Local Feature Maps

Having obtained three code maps *pcCode*, *oriCode*, and *phaCode* for each image, the next issue is how to match them for recognition. Since phase congruency is a dimensionless measure, we can use the absolute difference to measure the distance between two *pcCode* maps. Specifically, given two phase congruency code maps, *pcCode*1 and *pcCode*2, we define their normalized matching distance as

$$pcD = \frac{\sum\sum abs\left(pcCode1(\mathbf{x}) - pcCode2(\mathbf{x})\right)}{(L-1)\cdot S} \tag{5-22}$$

where S is the area of the image.

For comparing two orientation code maps, *oriCode*1 and *oriCode*2, we resort to the normalized angular distance proposed in [51], which is defined as

$$\begin{aligned} oriD &= \frac{\sum\sum ang\left(oriCode1(\mathbf{x}), oriCode2(\mathbf{x})\right)}{S\cdot J/2} \\ ang(p,q) &= \begin{cases} \min\left(p-q, q-p+J\right), & p \geq q \\ \min\left(q-p, p-q+J\right), & p < q \end{cases} \end{aligned} \tag{5-23}$$

When matching two phase code maps, we use a similar method as matching two orientation code maps. The matching distance between two phase code maps, *phaCode*1 and *phaCode*2, is given by

$$phaD = \frac{\sum\sum ang\left(phaCode1(\mathbf{x}), phaCode2(\mathbf{x})\right)}{S\cdot M/2} \tag{5-24}$$

5.2 Global Features

5.2.1 From Local to Global

In section 5.1, Gabor transforms are utilized to extract local orientation information. Actually, the Gabor transform can be regarded as a windowed Fourier transform. The corresponding Gabor transform (i.e. filtering) of a function f with respect to a local window function g is [157]

$$G[f](\mathbf{w},\mathbf{t}) = \int_{-\infty}^{+\infty} \dots \int_{-\infty}^{+\infty} f(\mathbf{x}) g(\mathbf{x}-\mathbf{t}) e^{-i(\mathbf{w}\cdot\mathbf{x})} d\mathbf{x} \qquad (5\text{-}25)$$

where $\mathbf{t}$, $\mathbf{w}$, $\mathbf{x} \in \mathrm{R}^n$, $d\mathbf{x} = dx_1 dx_2 \dots dx_n$, $\mathbf{x} = (x_1, x_2, \dots, x_n)$ and $\mathbf{w}\cdot\mathbf{x} = \sum_{k=1}^{n} w_k x_k$. The signal $f(\mathbf{t})$ to be analyzed is defined in the n-D spatial domain. $\mathbf{t}$ is the coordinate variable in the n-D spatial domain and correspondingly, $\mathbf{w}$ is the coordinate variable in the n-D frequency domain. The Gabor transform of f, $G[f](\mathbf{w}, \mathbf{t})$ can give the frequency spectrum of f for a specified frequency $\mathbf{w}$ at a specified position $\mathbf{t}$. For the convenience of discussion, we confine ourselves to the case that $n = 1$, g is a Gaussian-shaped window, and $f(x)$ is of finite length $[0, T]$. Then, the Gabor transform of f is

$$G[f](\omega, x) = \int_0^T e^{\frac{-(\tau-x)^2}{2\sigma^2}} f(\tau) e^{-i\omega\tau} d\tau, x \in [0, T] \qquad (5\text{-}26)$$

The parameter σ controls the size of the local window and the scale of the Gabor transform. Naturally, when σ goes to infinity, the whole signal $f(x)$, $x \in [0, T]$, is involved in calculating $G[f](\omega, x)$ and Eq. (5-26) is reduced to

$$G[f](\omega) = \int_0^T f(\tau) e^{-i\omega\tau} d\tau \qquad (5\text{-}27)$$

It is seen that $G[f]$ does not depend on x anymore, which implies that we lose the local information in the Gabor transform. Obviously, Eq. (5-27) is the Fourier transform of f.

The above discussion on the 1D case can be easily extended to the 2D case. For 2D images, by increasing the scale of the Gabor filters, more and more global information will be involved, yet the characterization of image local structures will be rapidly weakened. Particularly, if the scale of the Gabor filter goes to infinity, the Gabor transform will degrade to the 2D Fourier transform of the whole image. In such case, though the local characterization is totally lost, we can get the finest frequency resolution for the image analysis. Therefore, in our work the Fourier transform is selected as the global feature extractor.

5.2.2 Global Feature Matching

Now that the Fourier transform coefficients are used as the global feature, the next problem is how to measure the similarity of two given Fourier transforms. To this end, we can adopt the band-limited phase-only correlation (BLPOC) technique introduced in Chapter 4.

5.3 Local-global based Feature Extraction and Matching

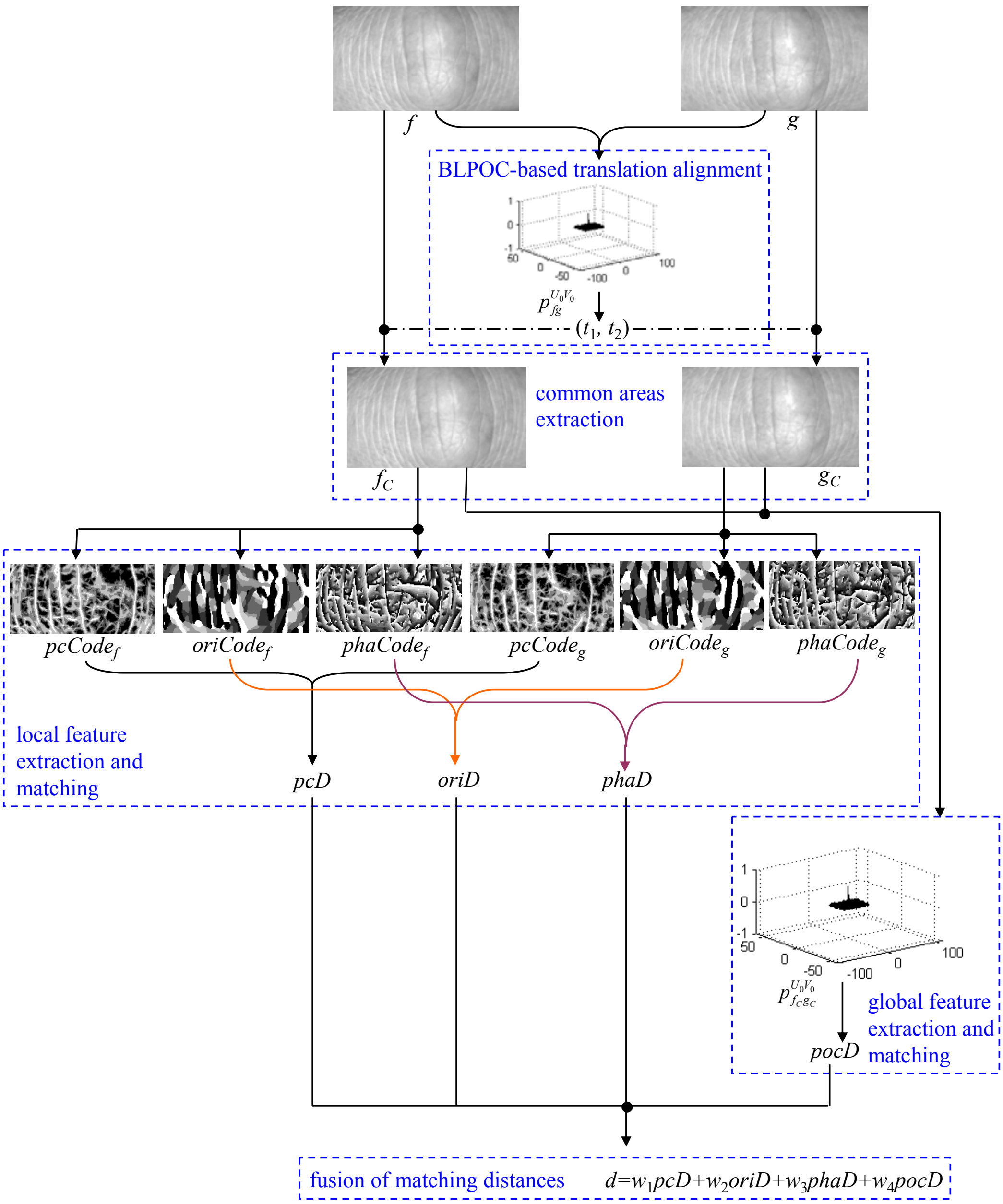


Figure 5.2 Illustration for the matching distance computation between a pair of FKP ROI images with our local-global feature extraction and fusion scheme.

In this section, we present our local-global based feature extraction and matching algorithm for FKP recognition. The entire process of our algorithm is illustrated using a pair of FKP images in Figure 5.2. Given two FKP ROI images f and g, the following four steps will be taken to compute their similarity.

Step 1*: *Translation alignment by global features with BLPOC

Although the FKP image acquisition device and the ROI extraction algorithm can reduce the geometric transformations between intra-class ROIs much, it is still inevitable that there is some displacement between intra-class ROIs. This will weaken the genuine matching scores. In our previous coding-based works [96, 98], this problem was addressed by translating one set of features in horizontal and vertical directions several times and the minimum of the resulting matching distances was considered to be the final matching distance. Here we solve this problem in a different way by evaluating the translation parameters between two ROIs using the BLPOC function. Then we crop the common regions, based on which the feature matching is performed.

The translation parameters (t_1, t_2) between f and g can be estimated from the peak location of the global BLPOC of them. Then, we can align f and g based on (t_1, t_2) and extract the common regions f_C and g_C. It should be noted that in our system, we will check the ratio between the common region area and the area of the original ROI. If $area(f_C)/area(f) < t$ (or $area(g_C)/area(g) < t$), where t is a threshold, f_C and g_C will be simply set as f and g. Generally, this will happen when the two FKP images are from different classes.

Step* 2: *Local feature extraction and matching

After alignment and common area cropping, three local feature maps are constructed from f_C and g_C each. Then, by matching the corresponding local feature maps, we could get three matching distances, *pcD*, *oriD*, and *phaD*. For technical details at this step, please refer to Section 5.1.

Step* 3: *Global feature extraction and matching

We use the peak value of the BLPOC function $p_{f_C g_C}^{U_0 V_0}$ between f_C and g_C to measure the similarity of their Fourier transform coefficients. Denote by *pocS* the peak value of $p_{f_C g_C}^{U_0 V_0}$, then the matching distance is defined as: $pocD = 1 - pocS$.

Step 4*: *Fusion of matching distances

Until now, four matching distances *pcD*, *oriD*, *phaD*, and *pocD* have been obtained. These four distances can be fused together to get the final matching distance. There are a couple of rules for the fusion of matching distances, such as the Simple-Sum (SS) rule, the MIn-Score (MIS) rule, the MAx-Score (MAS) rule, and the Matcher-Weighting (MW) rule [158]. In our case, *pcD*, *oriD*, *phaD*, and *pocD* can be considered to be obtained from four different matchers and we adopt the MW rule. With the MW fusion rule, weights are assigned according to the Equal Error Rate (EER) obtained on a training dataset by different matchers. Denote by e_k the EER of the matcher k, $k = 1,\ldots, 4$. Then, the weight w_k associated with matcher k can be calculated as

$$w_k = \left(1 / \sum_{k=1}^{4} \frac{1}{e_k}\right) / e_k \tag{5-28}$$

where $0 \le w_k \le 1$ and $\sum_{k=1}^{4} w_k = 1$. It is obvious that the weights are inversely proportional to the corresponding EERs. Then, the final matching distance is calculated as

$$d = w_1 pcD + w_2 oriD + w_3 phaD + w_4 pocD \tag{5-29}$$

5.4 Experiments

In order to demonstrate the efficacy of the proposed local-global feature extraction and matching scheme for FKP verification, experiments were conducted on the collected benchmark FKP database.

5.4.1 Determination of Parameters

In implementation, with respect to the quadrature pair filters, we utilized the log-Gabor filters whose transfer function in the frequency domain is

$$G_2(\omega,\theta_j) = \exp\left(-\frac{\left(\log\left(\omega/\omega_0\right)\right)^2}{2\sigma_r^2}\right)\cdot\exp\left(-\frac{\left(\theta-\theta_j\right)^2}{2\sigma_\theta^{\ 2}}\right) \tag{5-30}$$

where ω_0 is the filter's center frequency, σ_r controls the filter's radial bandwidth and σ_θ determines the filter's angular bandwidth. Parameters were empirically tuned based on a sub-dataset which contained the first 200 classes. Parameter settings used in our experiments are listed in Table 5.1, where ω_0^1, ω_0^2, and ω_0^3 represents the three center frequencies of the log-Gabor filters at three scales.

Table 5.1 Parameter settings for FKP verification.

n	J	σ_θ	L	ς	M	σ_r	ω_0^1	ω_0^2	ω_0^3	U_0/M_0	V_0/N_0
3	6	0.44	5	3	8	0.60	0.167	0.083	0.042	0.25	0.20

5.4.2 FKP Verification Results

In experiments, we gave the verification accuracy by using each single feature defined in section 5.1 and 5.2. To testify the efficacy of the proposed local-global-based matching scheme, we gave results by five fusion schemes: fusing all the three local features (F1), fusing the global feature with each of the three local features (F2 to F4), and fusing all the four features (F5). The verification results by the other proposed FKP recognition methods, including ImCompCode&MagCode, RCode2 (refer to Chapter 3), and BLPOC (refer to Chapter 4), are reported for comparison.

With respect to the performance measures, besides EER and ROC curves, we also use the decidability index d' [25].

Verification results obtained by using each single feature and different fusion schemes are summarized in Tables 5.2 and 5.3 respectively. Table 5.4 compares the proposed method F5 with the other FKP verification methods introduced in Chapters 3 and 4, RCode2, ImCompCode&MagCode, and BLPOC, and their ROC curves are shown in Figure 5.3. Distance distributions of genuine matchings and imposter machings obtained by the scheme F5 are plotted in Figure 5.4. Speed comparison of different FKP verification methods is listed in Table 5.5.

Table 5.2 Performance of each single feature for FKP verification.

feature type	EER (%)	*d'*
local orientation	1.66	4.2847
local phase	3.01	2.9213
local phase congruency	2.59	3.3811
Fourier transform	1.68	2.4745

Table 5.3 FKP verification by different feature fusion schemes.

Index	Fusion scheme	EER(%)	*d'*
F1	$w_1oriD + w_2pcD + w_3phaD$	1.37	4.3121
F2	$w_2oriD + w_4pocD$	0.44	4.5208
F3	$w_1pcD + w_4pocD$	0.63	4.4514
F4	$w_3phaD + w_4pocD$	0.83	4.3323
F5	$w_1oriD+w_2pcD+w_3phaD+w_4pocD$	0.36	4.7001

Table 5.4 Comparison of different FKP verification methods

Method	EER (%)	*d'*
RCode2	1.610	4.2835
BLPOC	1.676	2.4745
ImCompCode&MagCode	1.475	4.3224
F5	**0.358**	**4.7001**

Table 5.5 Speed comparison of different FKP verification methods

Method	Feature Extraction Time (msec) for 1 FKP Image	Time for 1 Matching (msec)
RCode2	17.6	1.2
BLPOC	1.4	2.1
ImCompCode&MagCode	105	1.6
F5	410	4.6

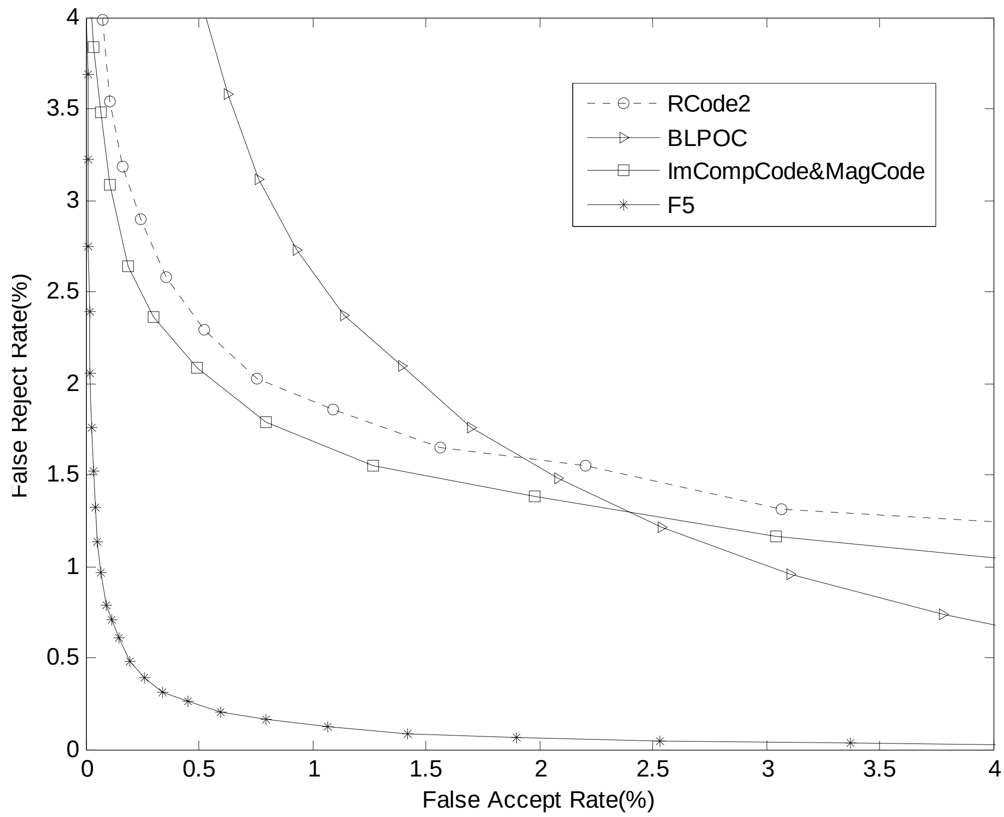


Figure 5.3 ROC curves obtained by the four FKP recognition methods.

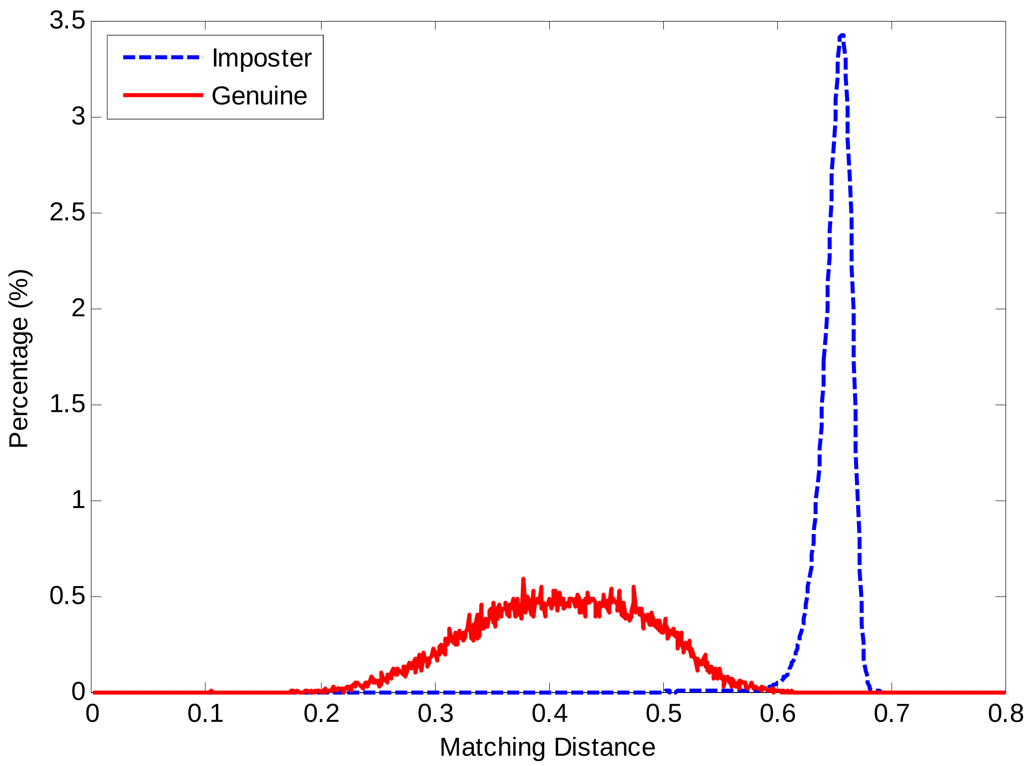


Figure 5.4 Distance distributions of genuine matchings and imposter matchings with the proposed scheme F5.

5.4.3 Palmprint Verification Results

Although the local-global feature extraction scheme introduced in this chapter is originally designed for the FKP verification, in fact, it can also be applied for the other kinds of biometric images. To corroborate this claim, in this sub-section we evaluate the performance of the proposed scheme for palmprint verification.

We use the PolyU palmprint database in experiments [159]. The PolyU palmprint database contains 7,752 grayscale palmprint images collected from 386 different palms. Each image is of the size 384×284 pixels. For each palm, there are around 20 samples collected in two sessions, where around 10 samples were collected each session. The average time interval between the first and second sessions was about 2 months. An ROI extraction procedure similar to that in [47] was used to extract the palmprint ROI of the size 128×128 pixels. In our experiments, we took images collected in the first session as the gallery set and images collected at the second session as the probe set. To obtain statistical results, each image in the probe set was matched with all the images in the gallery set. Under our experimental settings, the gallery set contains 3,889 images while the probe set contains 3,863 images. The numbers of genuine matchings and imposter matchings were 38,924 and 14,984,283, respectively.

The related parameter settings used are summarized in Table 5.6. Local feature maps extracted for a sample palmprint image are shown in Figure 5.5.

Table 5.6 Parameter settings for palmprint verification.

n	J	σ_θ	L	ς	M	σ_r	ω_0^1	ω_0^2	ω_0^3	U_0/M_0	V_0/N_0
3	6	0.44	5	3	8	0.60	0.167	0.083	0.042	0.25	0.20

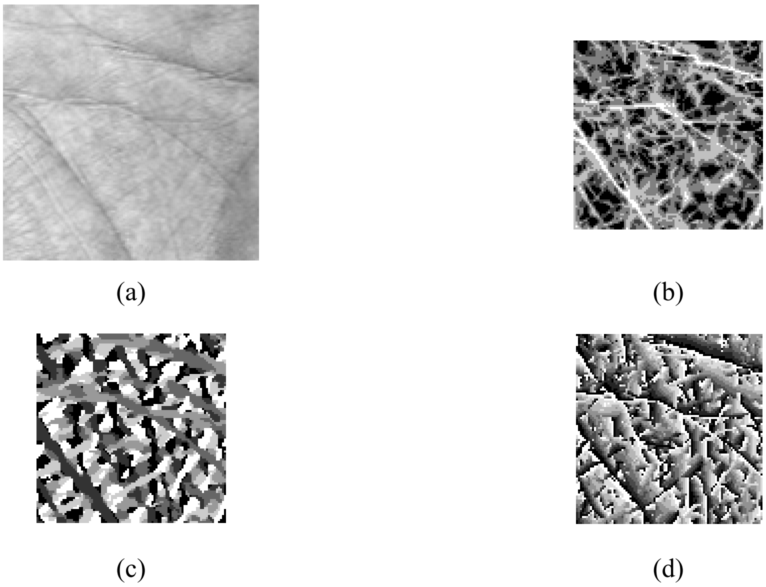


Figure 5.5 Local feature maps of a sample palmprint image. (a) is the original palmprint ROI image; (b) is its *pcCode* map; (c) is its *oriCode* map; (d) is its *phaCode* map.

Verification results by using each single feature are summarized in Table 5.7. The results by different feature fusion schemes are shown in Table 5.8. In Table 5.9, three state-of-the-art palmprint recognition methods, CompCode [51], RLOC [59] and OrdinalCode [53] were employed in comparison with the proposed local-global-based scheme F5 and their ROC curves are shown in Figure 5.6. Distance distributions of genuine matchings and imposter machings obtained by the scheme F5 are plotted in Figure 5.7.

Table 5.7 Palmprint verification by each single feature.

feature type	EER (%)	*d'*
local orientation	0.0673	7.0557
local phase	0.3417	5.4395
local phase congruency	1.6939	4.2120
Fourier transform	0.2434	6.0735

Table 5.8 Palmprint verification by different feature fusion.

Index	Fusion scheme	EER(%)	d'
F1	$w_1oriD + w_2pcD + w_3phaD$	0.0652	7.057
F2	$w_2oriD + w_4pocD$	0.0478	7.140
F3	$w_1pcD + w_4pocD$	0.2060	6.093
F4	$w_3phaD + w_4pocD$	0.1013	6.974
F5	$w_1oriD+w_2pcD+w_3phaD+w_4pocD$	0.0357	7.261

Table 5.9 Comparison of different palmprint verification methods.

Method	EER (%)	d'
CompCode	0.0575	6.3975
RLOC	0.1093	6.5422
OridinalCode	0.0916	6.4541
F5	**0.0357**	**7.2605**

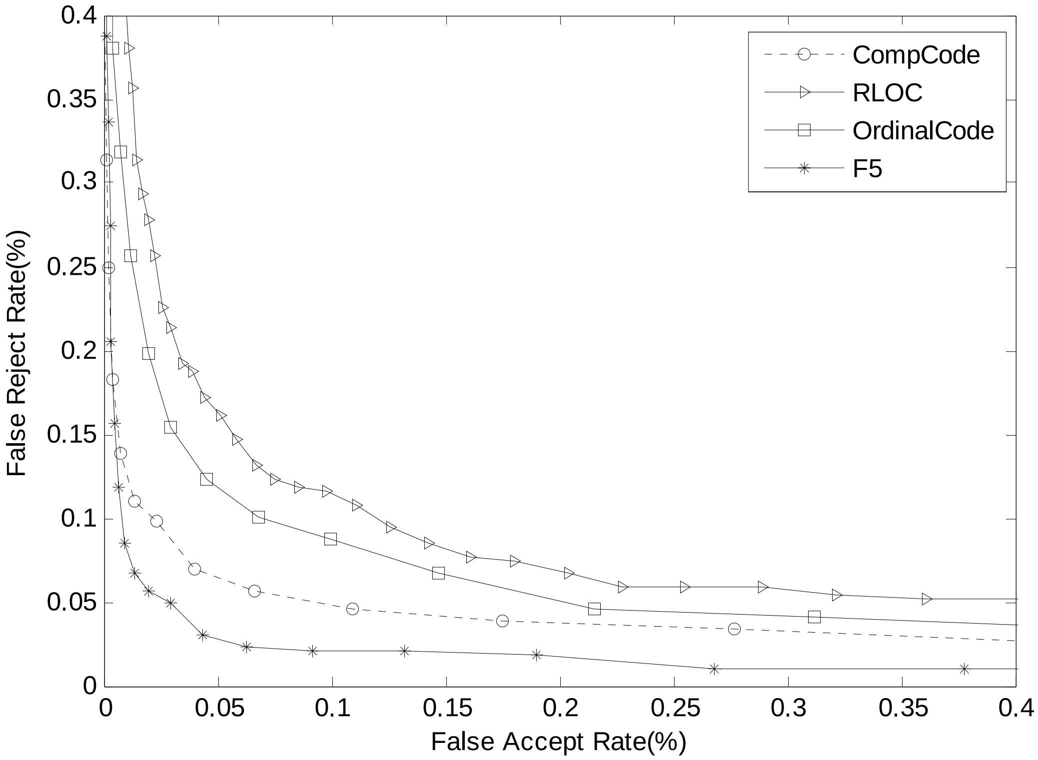


Figure 5.6 ROC curves obtained by four palmprint recognition methods.

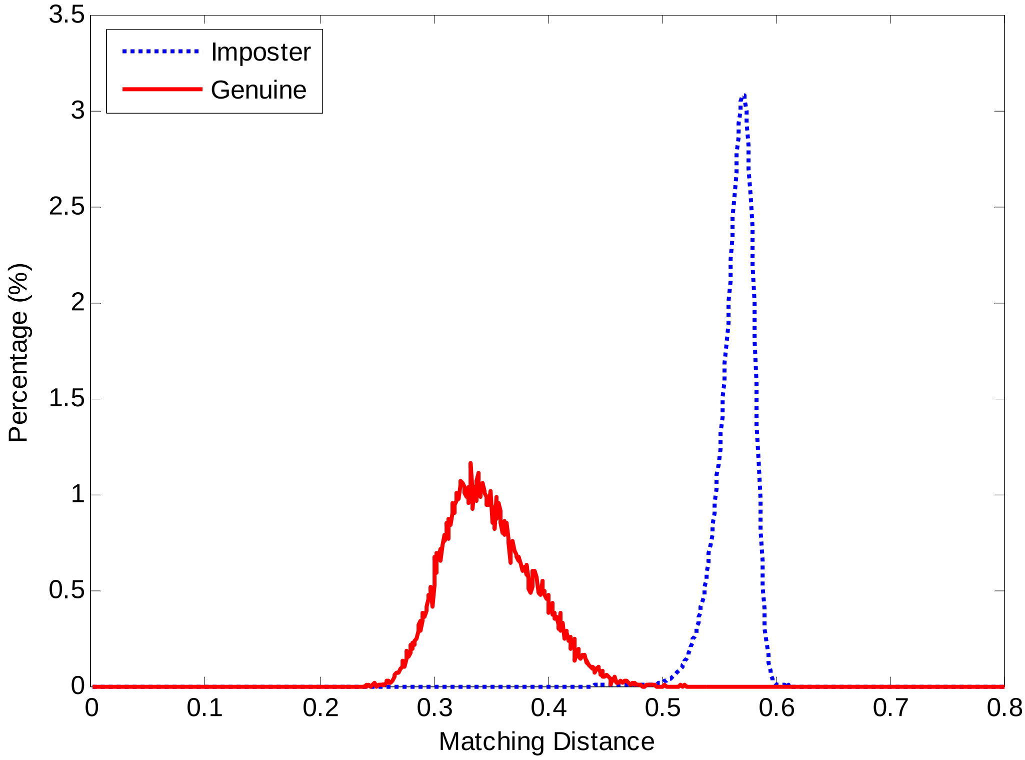


Figure 5.7 Distance distributions of genuine matchings and imposter matchings with the proposed scheme F5.

5.4.4 Discussions

From the results listed in Tables 5.2, 5.3, and 5.4, and the ROC curves shown in Figure 5.3, we can have the following findings: 1) the fusion of features perform better than using them separately, especially when the features are from different categories (global or local); 2) the best verification results can be obtained by the fusion scheme F5, which fuses the three local features and the one global feature; and 3) the fusion scheme F5 performs significantly better in terms of the verification accuracy than the other FKP verification methods evaluated, including RCode2, ImCompCode&MagCode, and BLPOC. As stated, RCode2, ImCompCode&MagCode can be classified as local-based methods. By contrast, in BLPOC, the Fourier transform of the whole image is taken as the feature so it is actually a global-based method. Therefore, the experimental results also corroborate the claim that methods fusing local and global information together can outperform the methods depending on only a specific kind of features, local or global. The above findings can also be reflected from experimental results on palmprint verification. Particularly, F5 performs better than the other state-of-the-art palmprint verification approaches.

It should be noted that the computation cost for the feature extraction and matching using the fusion scheme F5 is higher than the other methods evaluated. In a real system, maybe it is not necessary to incorporate all the three local features. The designer can choose only one or two of them in order to reduce the computation cost while keeping the verification accuracy meeting the requirement.

5.4 Summary

In this chapter, we proposed a novel framework to extract and combine the local and global features to improve the verification accuracy of the FKP-based biometric system. It is based on the fact that both local and global features are crucial for the image recognition and perception and they play different and complementary roles in such a process. For local features, we proposed to use the local orientation, the local phase, and the local phase congruency and presented an efficient method to compute them under the phase congruency computation framework. It was shown that Fourier transform is a natural choice for global feature extraction from the viewpoint of the time-frequency analysis. The proposed scheme exploits both local and global information for FKP verification, where the global features are also used to refine the alignment of FKP images in matching. To demonstrate the effectiveness of the proposed scheme, extensive experiments were performed on the benchmark FKP database. Experimental results indicated that the proposed local-global feature extraction and fusion scheme can lead to much better performance than the other feature extraction and matching methods for FKP verification. The newly proposed scheme can also be applied to other kinds of biometric images, such as palmprint.

Chapter 6. Embedded FKP Recognition System

In Chapters 2, 3, 4, and 5, we have discussed different aspects of the proposed FKP recognition prototype system, including the image acquisition device, the dataset collection, the image preprocessing, the ROI extraction, and the feature extraction and matching methods. However, in that prototype system, after the image is acquired, all the afterwards processing is performed on a general-purpose computer. Thus, it may have some drawbacks limiting its use in practice. At first, the overall size of the whole system is relatively large since usually a general-purpose computer has a large size. Secondly, the cost of whole system is high. To get rid of these disadvantages, in this chapter, we will present an embedded standalone FKP identification system, which will not depend on computer anymore. Such a system has a very small size and it costs much less than the originally developed prototype system. Hence, it can be used in practice readily.

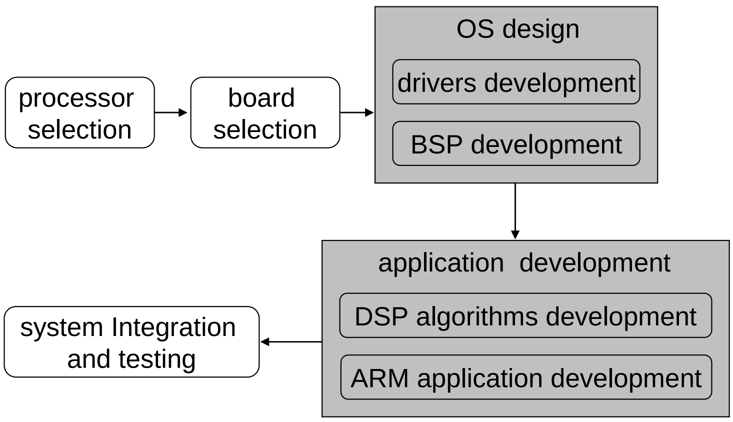


Figure 6.1 The general process for developing an embedded system.

Usually, as illustrated in Figure 6.1, the development of an embedded system consists of the following major steps: processor selection, board selection, OS (operating system) runtime image generation, application development, and system integration and testing. The hardware and software development involved in developing such an embedded FKP recognition system will be presented in detail in this chapter.

6.1 Hardware Development

6.1.1 Processor

For the processor, we adopt the OMAP3530 produced by the company Texas Instruments, USA. It is a dual-core processor, embedding with a 600MHZ ARM Cortex-A8 core and a 430MHZ TMS320C 64+ DSP core. Such a higher performance application processor is based on the enhanced OMAP™ 3 architecture. The OMAP™ 3 architecture aims to provide best-in-class video, image, and graphics processing capabilities which are sufficient to support the following: streaming video, 3D mobile gaming, video conferencing, and high-resolution still image. Many internal subsystems designed to support high-speed and multi-thread processing applications are included in this sensor. In addition, OMAP3530 processor comprises state-of-the-art power-management techniques which are required for high-performance mobile products. Besides, the device also offers memory stacking feature using the package-on-package (POP) implementation. OMAP3530 processor supports high-level operating systems such as Windows CE, Linux, Symbian OS and Palm OS. Detailed introduction about the features and parameters of OMAP3530 can be found at [160].

6.1.2 Board

In fact, based on the selected processor, we could design the board by ourselves. However, in order to save the time and energy, we decide to choose a proper board from existing ones embedded with an OMAP3530 processor in the market. Many factors, such as interfaces provided, the development kit provided, the size, and the price, need to be considered carefully when selecting a board for an application. In the end, we choose Devkit8000 manufactured by the Embest company for our system.

Devkit8000 Evaluation Board [161] is a compact board embedded with an OMAP3530 microprocessor. It fully explores all the features of this processor and supports up to

256MByte DDR SDRAM and 256MByte NAND Flash as well as high-speed USB2.0 OTG function. The board exposes many other hardware interfaces including RS232 serial port, LCD/TSP, DVI-D, S-Video, Ethernet, SD/MMC, keyboard, camera, SPI, I2C and JTAG. The user can boot the board from either SD card or NAND flash. The board is able to support WinCE and Linux OS. The outlook of the Devkit8000 board is shown in Figure 6.2. The board is of the size 110mm×95mm.

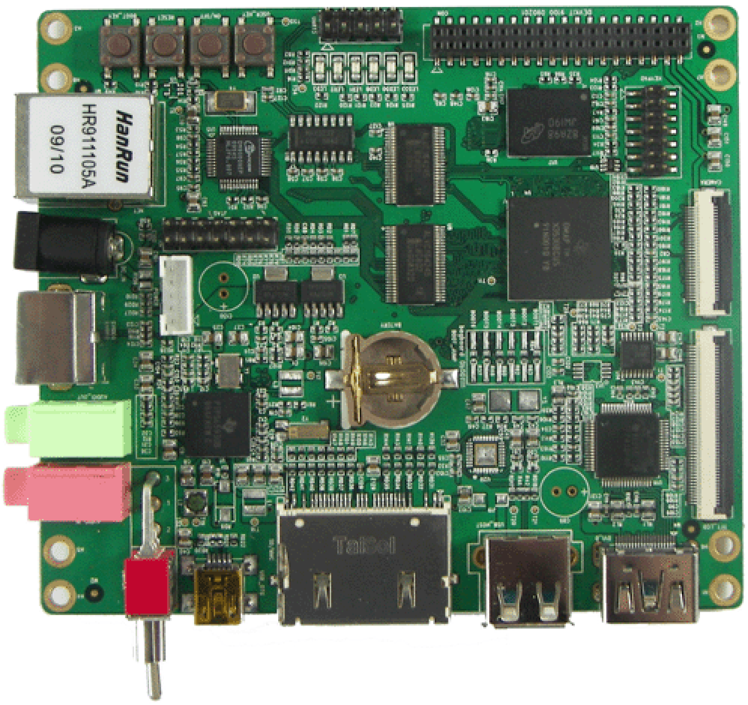
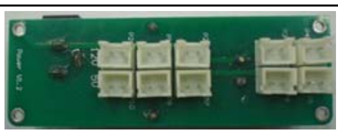
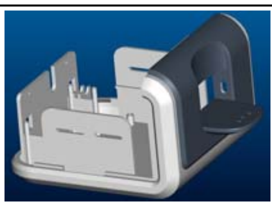


Figure 6.2 The outlook of the Devkit8000 board [161].

6.1.3 Peripherals

In addition to the main board, there are some other additional hardware components that are needed in our system. These components along with their photos and functionalities are summarized in Table 6.1.

Table 6.1 Additional components along with their photos and functionalities used in our system.

component	sample picture	functionality
TFT LCD and touch screen		It is used as a display; it is also used for the interaction between the user and the system.
CCD camera		It is used to capture FKP images.
lens		It is used in capturing FKP images.
LED light		It provides light when capturing images.
infrared sensor		It controls the light source. When the sensor detects finger, the LED light is turned on; otherwise, the LED light is turned off.
glass sheet		It prevents the device inside from the dust.
CAM8000-A		It is a camera module designed for Devkit8000.
Power module		It provides power for all the other components.
SD card		It is used to store FKP features or images.
shell		It encapsulates all the other components.

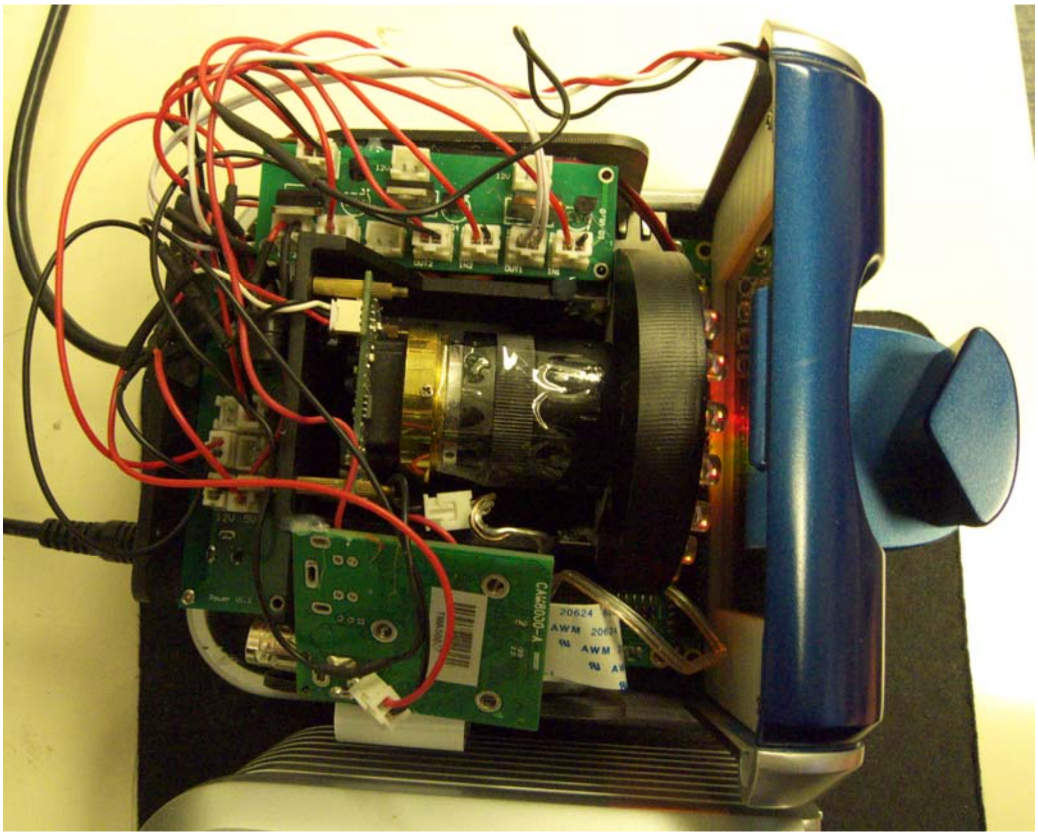


Figure 6.3 The inside of the FKP recognition system.

Figure 6.4 The outlook of the system when all the components have been assembled.

When all the hardware components are ready, they are assembled together to form the FKP recognition device. Figure 6.3 shows the inside of the device and Figure 6.4 shows the outlook of the device when all the components have been assembled. The overall size of the device is 165mm×138mm×103mm.

6.2 Software Development

After the hardware platform has been constructed, the next step is to develop the associated software for our system. Details about our embedded FKP recognition software development are introduced in this sub-section.

6.2.1 Building WinCE 6.0 Runtime Image

The first step to develop the software for an embedded device is to build an embedded operating system (OS) runtime image. In our system, we choose WinCE 6.0 as the embedded OS. WinCE is a 32-bit, native, and real-time operating system developed by Microsoft company to address the needs of handheld, mobile, and embedded devices. With support for multiple processor architectures, WinCE can be adapted to a variety of devices, such as smart-phones, pocket-pcs, thin-client terminals, digital cameras, point-of-scale, point-of-information, home and building automation, and human-machine interfaces [162, 163].

The critical step to generate the WinCE runtime image is to develop the board support package (BSP) for the target device. BSP is a common name referring to a software library required to load and run the operating system on a supported hardware device. BSP development typically involves the following: developing the boot loader, developing the OEM adaptation layer, developing the device drivers, and developing the OS runtime image configuration files. Usually, the hardware manufactory will provide a BSP template for the customer. Then, the customer could build a customized BSP based on the provided template. For details about BSP development for WinCE, please refer to [163].

Table 6.2 Software tools used in developing WinCE runtime image.

Visual Studio 2005
Windows Embedded CE 6.0
Visual Studio 2005 SP1
Windows Embedded CE 6.0 SP1
Windows Embedded CE 6.0 R2
WinCEPB60-081231-Product-Update.msi
.Net Compact Framework 2.0 SP2
Active Sync 4.5

To build the WinCE runtime image, the software tools listed in Table 6.2 should be installed on the development computer.

After the WinCE runtime image is generated, we download it to the Nand flash of Devkit8000.

6.2.2 GPP+DSP Application Development

Having the WinCE runtime image created, the following step is to develop the FKP recognition software that could execute on WinCE. Using the same VS2005 IED environment to develop applications, the application development process is quite similar to desktop Windows. In our case, we implement the software with Visual C++ 2005. Due to the limited hardware resources, developers should be conscious of the following when developing WinCE applications: 1) the target device may has limited system memory and limited storage as well as a slow processor; 2) developers need to release resources that are not needed timely; 3) the target device may not have display, mouse, or keyboard; 4) the target device may have a display with very low resolution; and 5) power to the device may be turned off unexpectedly [163].

As stated in section 6.1.1, the OMAP3530 processor has two processors, one general-purpose processor (ARM), and one digital signal processor (DSP). When powered on, the WinCE operating system and the WinCE based applications actually run on the general purpose processor (GPP). As we know, DSP has a strong signal processing capability. Thus, in order to fully make use of the computation power of OMAP3530, we need to assign some workload to the DSP side. Here come the problems: how to assign workload to the DSP and how the application running on the GPP can communicate with algorithms running the DSP.

To facilitate the GPP+DSP application development, TI company has provided developers a software framework, namely Codec Engine [164-166]. From the application developer's view, the Codec Engine is a set of APIs that can be utilized to instantiate and run DSP algorithms. The Codec Engine is easy to use, extensible and configurable. The application code running on the GPP calls the Codec Engine APIs. Within the Codec Engine, stubs and skeletons are used by the APIs to access the core engine and the actual codecs, which can run on the DSP. Figure 6.5 shows the overall blue chart of an application that uses the Codec Engine [164]. It also shows different user roles in creating various parts of the application.

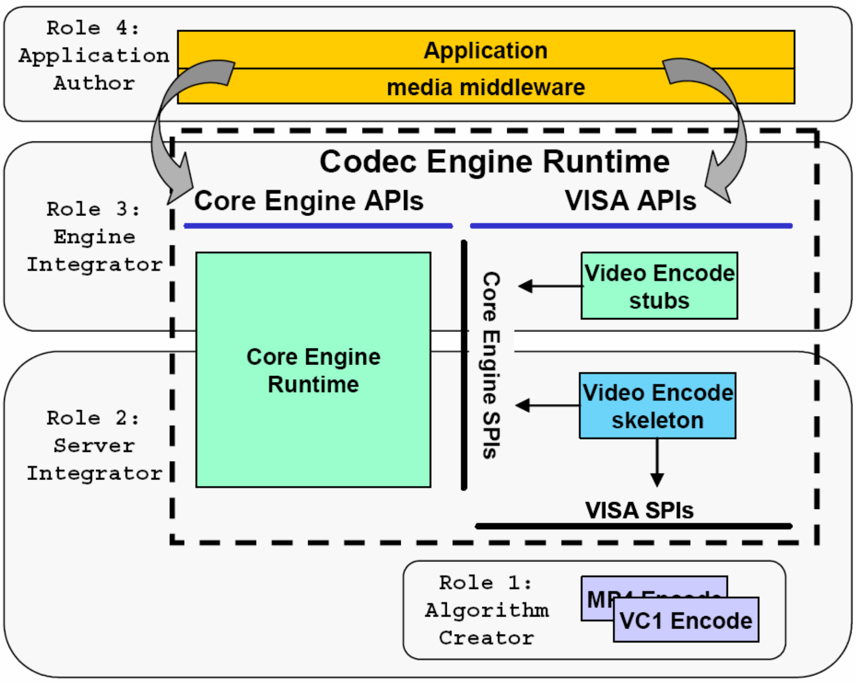


Figure 6.5 Architecture of an application based on the Codec Engine [164].

Table 6.3 Different user roles and their tasks when using Codec Engine to develop an application.

user role	task
Algorithm Creator	The Algorithm Creator is responsible for creating DSP algorithms and providing the necessary encapsulating to enable these algorithms to be consumed and configured by Codec Engine. The output of an Algorithm Creator is a released Codec package, which will be used by the Server Integrator.
Server Integrator	In order to support Engines with remote codecs, a Codec Server needs to be created. The Codec Server integrates and configures various components that are necessary to house the codecs and generates an executable image. The Server Integrator releases the DSP executable to the Engine Integrator.
Engine Integrator	Various Engine configurations are defined by the Engine Integrator. This includes the names of the Engines, the names of the codecs within each Engine, and the name of the Codec Server image.
Application Author	The Codec Engine APIs are utilized by the application author to create/delete preconfigured Engine instances, manipulate codecs, and apply buffers appropriate for the codecs, etc. Because the Codec Engine cannot process any I/O, the application should provide I/O interfaces. Besides, the Application Author is responsible for implementing functionalities of the application and for generating the final executable image.

As it can be seen from Figure 6.5, using the Codec Engine as a framework to develop GPP+DSP applications, several different user roles will be involved [164-166]. Their tasks are summarized and listed in Table 6.3.

To develop Codec Engine based applications, the software tools listed in Table 6.4 are required.

Table 6.4 Software tools used when developing Codec Engine based applications

Code Composer Studio
Code Generation Tools
DSP/BIOS
XDCTools
Active Perl
WinCE DVSDK

6.2.3 FKP Recognition Software

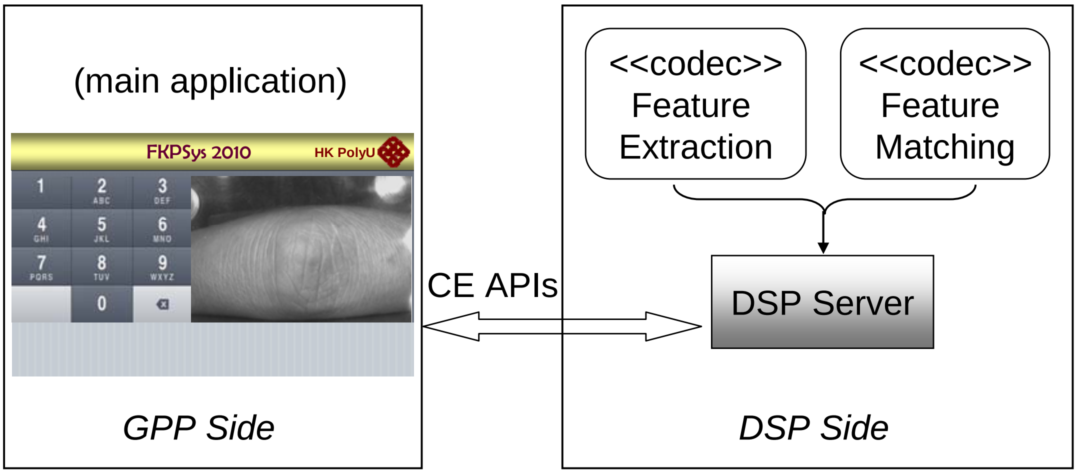


Figure 6.6 Architecture of the FKP recognition software.

Figure 6.7 Outlook of the FKP recognition system when the recognition software is running.

Figure 6.8 The FKP recognition system is working in the registration mode.

Our FKP recognition software adopts the GPP+DSP architecture and uses the Codec Engine framework. The main application runs on the GPP and it is responsible for capturing the FKP image, handling UI interactions, and handling I/O. Two codecs are created to encapsulate two key algorithms running on the DSP, feature extraction and feature matching. These two codecs are integrated into a codec server, which can be manipulated by the main application running on the GPP via Core Engine APIs. Such architecture is illustrated in Figure 6.6. In order to further improve the algorithms' performance, when implementing DSP codecs, we made use of two libraries provided by the TI company. One is TMS320C64+ DSP Library [167] and the other one is TMS320C64+ Image/Video Library [168]. All the procedures contained in these two libraries are optimized for the TMSC64+ DSP (DSP core used in OMAP3530). By using these routines, we could achieve execution speeds considerably faster than equivalent code written in standard ANSI C language.

When the development of the software is completed, the whole FKP recognition system is ready. Figure 6.7 shows the outlook of our embedded FKP recognition system with the recognition software running. In Figure 6.8, the system is working in the registration mode.

6.3 Summary

Based on the technologies developed from Chapters 2 to 5, in this chapter we present an embedded FKP recognition system. It is a standalone system and does not depend on a general-purpose computer anymore. OMAP3530 is chosen as the core processor and WinCE 6.0 is selected as the operating system for this system. The associated FKP recognition software is based on a GPP+DSP architecture which can fully explore the computation power of the OMAP3530 processor. Such a system has the merits of small size, fast speed, and cost effective so it can be readily used in practice. A video clip that shows how this system works can be found online at http://www.youtube.com/watch?v=HLSlj_PkWyc or at http://v.youku.com/v_show/id_XMjI5NDA4MzE2.html.

Chapter 7. Conclusion

This thesis has studied various aspects of developing a practical FKP-based personal authentication system. It corroborates that by proper exploration, finger-knuckle-print has a great potential to serve as a promising biometric identifier. In this chapter, at first, the verification accuracy of several widely used biometric identifiers will be summarized. Then, a summary of the contributions of this thesis will be given. Open problems and possible future works will also be discussed.

7.1 Verification Accuracy of Different Biometric Identifiers

Table 7.1 Verification accuracy of widely used biometric identifiers

Biometric Identifier	Dataset	Number of Classes	Number of Samples	EER (%)
Fingerprint	FVC2006	140	1680	2.155
Iris	CASIA-IrisV3-Interval	396	2655	0.004
	ICE2005-Left	120	1528	0.011
	ICE2005-Right	124	1425	0.006
Palmprint	PolyU	386	7752	0.038
Face	XM2VTS	295	1180	2.7
	FERET	70	420	4.6
Hand shape	Bogazici	458	1374	1.79
FKP	PolyU	660	7920	0.36

Although this thesis mainly deals with FKP recognition, we summarize the verification accuracy of several widely used biometric identifiers in Table 7.1 to give the reader a general impression of the development of the biometrics field. The reported biometric identifiers include fingerprint [169], iris [30], palmprint [62], face [170], and hand shape [70] and the results are extracted from some latest papers to our knowledge. It needs to be noted that you can not conclude which one is better and which one is worse only based on the results listed in Table 7.1 because the scales of the datasets and the experimental protocols are different from each other. Table 7.1 can only serve as a reference.

7.2 Contributions

The main contribution of this work is that a practical brand-new FKP-based personal authentication system was constructed. In achieving this overall objective, a number of technologies were developed and they can be summarized as follows:

- A specially designed FKP imaging device was produced. In our device, we make use of a triangular block to control the finger freedom. This gadget does not sacrifice the user convenience and it is easy to use. Such a design brings the following merits: 1) the acquisition device could be easily made to a small size; 2) image around the finger knuckle area is captured directly, which largely simplifies the following data preprocessing steps; and 3) since the finger knuckle is slightly bent when being captured, the distinctive FKP texture patterns can be clearly imaged, which makes the proposed FKP system have high accuracy.
- A large FKP database was established using the developed FKP image acquisition device. This database is intended to be a benchmark to evaluate the performance of various FKP recognition methods, and now it is publicly available at [109]. We collected samples in two separate sessions. In each session, the subject was asked to provide 6 images for each of the left index finger, the left middle finger, the right index finger and the right middle finger. Therefore, 48 images from 4 fingers were collected from each subject. In total, the database contains 7,920 images from 660 different fingers. This database is the largest of its kind currently.
- A novel efficient FKP ROI extraction algorithm is proposed based on the intrinsic characteristics of FKP images. Such an ROI extraction step is critical for the following feature extraction and matching.
- Performances of several coding-based feature extraction and matching methods were examined on the collected FKP database.

- A novel coding-based feature extraction and matching method, namely ImCompCode&MagCode was proposed, which is an extended version of CompCode. In ImCompCode&MagCode, not only the orientation information but also the magnitude information is considered.
- Two novel efficient and compact coding based feature extraction and matching approaches, RCode1 and RCode2, were proposed based on Riesz transforms, which have been attracting much attention from researchers in the engineering field. RCode2 is demonstrated to be the best 3-bit coding-based FKP recognition method.
- A novel global feature based FKP recognition scheme was proposed. In that approach, Fourier transform coefficients of images were taken as global features. For matching, the band-limited phase-only correlation technique was adopted.
- A novel local-global information combination based FKP recognition approach was proposed. It is inspired by the results of psychophysics and neurophysiology studies that both local and global information is crucial for the image perception and recognition of human beings and they play different but complementary roles. A global feature reflects the holistic characteristics of the image and is suitable for coarse representation, while a local feature encodes more detailed information within a specific local region and is appropriate for finer representation. We use the local orientation, the local phase, and the local phase congruency as the local features and they reflect different aspects of information within a local patch and none of them can be covered by the others. Fourier transform coefficients are naturally taken as global features. Such kinds of local and global features are naturally linked via the framework of time-frequency analysis. With the global Fourier features, the alignment between intra-class FKP ROIs can also be refined. Such a local-global fusion based feature extraction and matching scheme is effective not only for the FKP recognition but also for the recognition of some other kinds of biometric images, such as the palmprint.
- Based on the developed technologies, a standalone embedded FKP recognition system

was implemented. It is based on the OMAP3530 processor and the WinCE OS. It has the merits of high speed, small size, and cost effective. Such a system can be readily used in practice. It represents the first of its kind.

7.3 Future Work

It is hoped that this thesis has shown that there are many possibilities to be explored in the use of FKP for personal authentication. Several directions could extend the current work and improve the accuracy and robustness of the FKP recognition.

- Since FKP is relatively a new member in the biometrics family, to make it accepted, its reliability and uniqueness need to be validated on an even larger database. So in the near future, we plan to collect FKP samples from a fixed large group of people continuously.
- FKP samples in our current database are all from Chinese people. Thus, whether the findings and the associated technologies developed are applicable to other groups of people, e.g., persons from western countries, needs further investigation. To this end, we may need to extend our FKP database to include more samples from other nations, especially western nations.
- It should be noted that although we use a triangular block to control the finger freedom in FKP image acquisition, there are still variations for the same finger at different collection sessions. Sometimes such variations can result in severe affine transformationss or even non-elastic deformations among intra-class FKPs. As a result, feature maps of such FKPs can have large matching distances. Our current feature extraction and matching schemes developed could not address these problems very well. Hence, in the future, we will focus on how to deal with affine or even non-elastic deformations between FKP images from the same finger to further improve the system's performance.
- Our current FKP recognition system does not have the live-ness detection function. In

the future, we plan to add such a module to make the system have the capability of anti-spoofing.

Bibliography

[1] A.K. Jain, P.J. Flynn, and A. Ross, *Handbook of Biometrics*. Springer, 2007.

[2] D. Zhang, *Automated Biometrics-Technologies and Systems*. Kluwer Academic Publishers, 2000.

[3] W.S. Coats, A. Bagdasarian, T.J. Helou, and T. Lam, *The Practitioner's Guide to Biometrics*. American Bar Association, 2007.

[4] S. Nanavati, M. Thieme, and R. Nanavati, *Biometrics: Identity Verification in a Networked World*. John Wiley & Sons, Inc, 2002.

[5] A.A. Ross, K. Nandakumar, and A.K. Jain, *Handbook of Multibiometrics*. Springer, 2006.

[6] Biometrics, http://en.wikipedia.org/wiki/Biometrics.

[7] E-Channel, http://en.wikipedia.org/wiki/E-Channel.

[8] BCC Research, http://www.bccresearch.com/report/IFT042B.html.

[9] A.K. Jain, R. Bolle, and S. Pankanti, editors, *Biometrics: Personal Identification in Networked Society*. Kluwer Academic Publishers, 1999.

[10]K. Delac, M. K. Delac, and M. Grgic, *Face Recognition*. I-Tech Education and Publishing, 2007.

[11]H. Wechsler, *Reliable Face Recognition Methods - System Design, Implementation and Evaluation*. Springer, 2006.

[12]S.Z. Li and A.K. Jain, *Handbook of Face Recognition*. Springer, 2005.

[13]Y. Fang, T. Tan, and Y. Wang, "Fusion of global and local features for face verification", in *Proc. ICPR*, pp. 382-385, 2002.

[14]C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition", *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 467-476, 2002.

[15]A.M. Bronstein, M.M. Bronstein, and R. Kimmel, "Three-dimensional face recognition", *Int. J. Computer Vision*, vol. 64, no. 1, pp. 5-30, 2005.

[16]A.M. Bronstein, M.M. Bronstein, and R. Kimmel, "Expression-invariant representation of faces", *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 188-197, 2007.

[17]S. Gundimada and V.K. Asari, "A novel neighborhood defined feature selection on phase congruency images for recognition of faces with extreme variations", *Int. J. Information*

Technology, vol. 3, no. 1, pp. 25-31, 2007.

[18]L. Shen and L. Bai, “A review on Gabor wavelets for face recognition”, *Pattern Anal. Appl.*, vol. 9, no. 10, pp. 273-292, 2006.

[19]B. Zhang, S. Shan, X. Chen, and W. Gao, “Histogram of Gabor phase patterns (HGPP): a novel object representation approach for face recognition”, *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 57-68, 2007.

[20]J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, and Y. Ma, “Robust face recognition via sparse representation”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210-227, 2009.

[21]Y. Su, S. Shan, X. Chen, and W. Gao, “Hierarchical ensemble of global and local classifiers for face recognition”, *IEEE Trans. Image Process.*, vol. 18, no. 8, pp. 1885-1896, 2009.

[22]S. Xie, S. Shan, X. Chen, and J. Chen, “Fusing local patterns of Gabor magnitude and phase for face recognition”, *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1349-1361, 2010.

[23]J. Daugman, “High confidence visual recognition of persons by a test of statistical independence”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 11, pp. 1148-1161, 1993.

[24]J. Daugman, “How iris recognition works”, *IEEE Trans. Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 21-30, 2004.

[25]J. Daugman, “The importance of being random: statistical principles of iris recognition”, *Pattern Recognit.*, vol. 36, no. 2, pp. 279-291, 2003.

[26]Z. Sun, Y. Wang, T. Tan, and J. Cui, “Cascading statistical and structural classifiers for iris recognition”, in *Proc. ICPR*, pp. 1261-1262, 2004.

[27]Z. Sun, Y. Wang, T. Tan, and J. Cui, “Improving iris recognition accuracy via cascaded classifiers”, *IEEE Trans. Systems, Man, and Cybernetics-Part C*, vol. 35, no. 3, 435-441, 2005.

[28]X. Yuan and P. Shi, “Iris feature extraction using 2D phase congruency”, in *Proc. ICITA*, pp. 437-441, 2005.

[29]K. Miyazawa, K. Ito, T. Aoki, K. Kobayashi, and H. Nakajima, “An effective approach for iris recognition using phase-based image matching”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 10, pp. 1741-1756, 2008.

[30]Z. Sun and T. Tan, “Ordinal measures for iris recognition”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2211-2226, 2009.

[31]K.W. Bowyer, K. Hollingsworth, and P.J. Flynn, “Image understanding for iris biometrics: a survey”, *Computer Vision and Image Understanding*, vol. 110, no. 2, pp. 281-307, 2008.

[32]R. B. Hill, “Retinal identification”, in *Biometrics: Personal Identification in Networked Society*, A.K. Jain, R. Bolle, and S. Pankanti, Eds., Kluwer Academic, 1999.

[33]H. Borgen, P. Bours, and S.D. Wolthusen, “Visible-spectrum biometric retina recognition”, in *Proc. Int. Conf. Intelligent Information Hiding and Multimedia Signal Processing*, pp. 1056-1062, 2008.

[34]P. Yan and K.W. Bowyer, “Biometric recognition using 3D ear shape”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 8, pp. 1297-1308, 2007.

[35]B. Bhanu and H. Chen, *Human Ear Recognition*. Springer, 2008.

[36]D. Zhang, Z. Liu, J. Yan, and P. Shi, “Tongue-Print: a novel biometrics pattern”, in *Proc. ICB*, pp.1174-1183, 2007.

[37]D. Zhang, Z. Liu, and J. Yan, “Dynamic tongueprint: a novel biometric identifier”, *Pattern Recognit.*, vol. 43, no. 3, pp. 1071-1082, 2010.

[38]L. Hong, Y. Wan, and A. Jain, “Fingerprint image enhancement: algorithm and performance evaluation”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 777-789, Aug. 1998.

[39]D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*. Springer, 2003.

[40]N. Ratha and R. Bolle, *Automatic Fingerprint Recognition Systems.* Springer, 2004.

[41]J. Gu, J. Zhou, and C. Yang, “Fingerprint recognition by combining global structure and local cues”, *IEEE Trans. Image Process.*, vol. 15, no. 7, pp. 1952-1964, 2006.

[42]K. Ito, H. Nakajima, K. Kobayashi, T. Aoki, and T. Higuchi, “A fingerprint matching algorithm using phase-only correlation”, *IEICE Trans. Fundamentals*, vol. E87-A, no. 3, pp. 682-691, 2004.

[43]K. Ito, A. Morita, T. Aoki, H. Nakajima, K. Kobayashi, and T. Higuchi, “A fingerprint recognition algorithm combining phase-based image matching and feature-based matching”, in *Proc. ICB*, pp. 316-325, 2005.

[44]J. Zhang, Z. Ou, and H. Wei, “Fingerprint matching using phase-only correlation and Fourier-Mellin transforms”, in *Proc. ISDA*, pp. 379-383, 2006.

[45]Q. Zhao, D. Zhang, L. Zhang, and N. Luo, “Adaptive fingerprint pore modeling and extraction”, *Pattern Recognit.*, vol. 43, no. 8, pp. 2833-2844, 2010.

[46]Q. Zhao, D. Zhang, L. Zhang, and N. Luo, “High resolution fragmentary fingerprint alignment using pore-valley descriptors”, *Pattern Recognit.*, vol. 43, no. 3, pp. 1050-1061, 2010.

[47]D. Zhang, W. Kong, J. You, and M. Wong, “Online palmprint identification”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1041-1050, 2003.

[48]G. Lu, D. Zhang, and K. Wang, “Palmprint recognition using eigenpalms features”, *Pattern Recognit. Letters*, vol. 24, no. 9, pp. 1463-1467, 2003.

[49]X. Wu, D. Zhang, and K. Wang, "Fisherpalms based palmprint recognition", *Pattern Recognit. Letters*, vol. 24, no. 15, pp. 2829-2838, 2003.

[50]C. Han, H. Cheng, C. Lin, and K. Fan, "Personal authentication using palm-print features", *Pattern Recognit.*, vol. 36, no. 2, pp. 371-381, 2003.

[51]A. Kong and D. Zhang, "Competitive coding scheme for palmprint verification", in *Proc. ICPR*, pp. 520-523, 2004.

[52]L. Zhang and D. Zhang, "Characterization of palmprints by wavelet signatures via directional context modeling", *IEEE Trans. System, Man and Cybernetics B*, vol. 34, no. 3, pp. 1335-1347, 2004.

[53]Z. Sun, T. Tan, Y. Wang, and S.Z. Li, "Ordinal palmprint representation for personal identification", in *Proc. CVPR*, pp. 279-284, 2005.

[54]A. Kong, D. Zhang, and M. Kamel, "Palmprint identification using feature-level fusion", *Pattern Recognit*., vol. 39, no. 3, pp. 478-487, 2006.

[55]F. Li, M.K.H. Leung, and X. Yu, "A two-level matching scheme for speedy and accurate palmprint identification", in *Proc. Int. Multimedia Modeling Conference*, pp. 323-332, 2007.

[56]X. Pan, Q. Ruan, and Y. Wang, "Palmprint recognition using fusion of local and global features", in *Proc. Int. Symposium on Intelligent Signal Processing and Communication Systems*, pp. 642-645, 2007.

[57]Z. Guo, D. Zhang, L. Zhang, and W. Zuo, "Palmprint verification using binary orientation co-occurrence vector", *Pattern Recognit. Letters*, vol. 30, no. 13, pp. 1219-1227, 2009.

[58]D. Huang, W. Jia, and D. Zhang, "Palmprint verification based on principal lines", *Pattern Recognit.*, vol. 41, no. 4, pp. 1316-1328, 2008.

[59]W. Jia, D. Huang, and D. Zhang, "Palmprint verification based on robust line orientation code", *Pattern Recognit.*, vol. 41, no. 5, pp. 1504-1513, 2008.

[60]K. Ito, T. Aoki, H. Nakajima, K. Kobayashi, and T. Higuchi, "A palmprint recognition algorithm using phase-only correlation", *IEICE Trans. Fundamentals*, vol. E91-A, no. 4, pp. 1023-1030, 2008.

[61]V. Struc and N. Pavesic, "Phase congruency features for palm-print verification", *IET Signal Process.*, vol. 3, no. 4, pp. 258-268, 2009.

[62]A. Kong, D. Zhang, and M. Kamel, "A survey of palmprint recognition", *Pattern Recognit.*, vol. 42, no. 7, pp. 1408-1418, 2009.

[63]A.K. Jain and J. Feng, "Latent palmprint matching", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 6, pp. 1032-1047, 2009.

[64]D. Zhang, G. Lu, W. Li, L. Zhang, and N. Luo, "Palmprint recognition using 3-D information",

IEEE Trans. System, Man and Cybernetics C, vol. 39, no. 5, pp. 505-519, 2009.

[65]W. Li, L. Zhang, D. Zhang, G. Lu, and J. Yan, "Efficient joint 2D and 3D palmprint matching with alignment refinement", in *Proc. CVPR*, 2010.

[66]D. Zhang, Z. Guo, G. Lu, L. Zhang, and W. Zuo, "An online system of multi-spectral palmprint verification", *IEEE Trans. Instrumentation and Measurement*, vol. 59, no. 2, pp. 480-490, 2010.

[67]A.K. Jain, A. Ross, and S. Pankanti, "A prototype hand geometry-based verification system", in *Proc. AVBPA*, pp. 166–171, 1999.

[68]A.K. Jain and N. Duta, "Deformable matching of hand shapes for verification", in *Proc. ICIP*, pp. 857–861, 1999.

[69]R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos, "Biometric identification through hand geometry measurements", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1168-1171, 2000.

[70]N. Duta, "A survey of biometric technology based on hand shape", *Pattern Recognit.*, vol. 42, no. 11, pp. 2797-2806, 2009.

[71]Y. Ding, D. Zhuang, and K. Wang, "A study of hand vein recognition method", in *Proc. IEEE Int. Conf. Mechatronics & Automation*, pp. 2106-2110, 2006.

[72]J. Wang, W. Yau, A. Suwandy, and E. Sung, "Personal recognition by fusing palmprint and palm vein images based on "Laplacianpalm" representation", *Pattern Recognit.*, vol. 41, no. 5, pp. 1531-1544, 2008.

[73]Z. Liu, Y. Yin, H. Wang, S. Song, and Q. Li, "Finger vein recognition with manifold learning", *Journal of Network and Computer Applications*, vol. 33, no. 3, pp. 275-282, 2010.

[74]D. Mulyono and H.S. Jinn, "A study of finger vein biometric for personal identification", in *Proc. Int. Symposium on Biometrics and Security Technologies*, pp. 1-8, 2008.

[75]C. Wilson, *Vein Pattern Recognition*. CRC Press, 2010.

[76]D.L. Woodard and P.J. Flynn, "Personal identification utilizing finger surface features", in *Proc. CVPR*, pp. 1030-1036, 2005.

[77]D.L. Woodard and P.J. Flynn, "Finger surface as a biometric identifier", *Computer Vision and Image Understanding*, vol. 100, no. 3, pp. 357–384, 2005.

[78]D.L. Woodard, "Exploiting finger surface as a biometric identifier", PhD Thesis, University of Notre Dame, 2004.

[79]C. Ravikanth and A. Kumar, "Biometric authentication using finger-back surface", in *Proc. CVPR*, pp. 1-6, 2007.

[80]A. Kumar and C. Ravikanth, "Personal authentication using finger knuckle surface", *IEEE*

Trans. Information Forensics and Security, vol. 4, no. 1, pp. 98-109, 2009.

[81]A. Kumar and Y. Zhou, "Human identification using knucklecodes", in *Proc. BTAS*, 2009.

[82]A. Kumar and Y. Zhou, "Personal identification using finger knuckle orientation features", *Electronic Letters*, vol. 45, no. 20, pp. 1023-1025, 2009.

[83]Q. Li, Z. Qiu, D. Sun, and J. Wu, "Personal identification using knuckleprint", in *Proc. SinoBiometrics*, pp. 680-689, 2004.

[84]L. Nanni and A. Lumini, "A multi-matcher system based on knuckle-based features", *Neural Computing & Applications*, vol. 18, no. 1, pp. 87-91, 2009.

[85]K. Nakajima, Y. Mizukami, K. Tanaka, and T. Tamura, "Footprint-based personal authentication", *IEEE Trans. Biomedical Engineering*, vol. 47, no. 11, pp. 1534-1537, 2000.

[86]J.W. Jung, Z. Bien, S.W. Lee, and T. Sato, "Dynamic-footprint based person identification using mat-type pressure sensor", in *Proc. IEEE 25th Annual Int. Conf. Engineering in Medicine and Biology Society*, pp. 2937-2940, 2003.

[87]W. Jia, J. Gui, R. Hu, Y. Lei, and X. Xiao, "Newborn footprint recognition using subspace learning methods", in *Proc. ICIC*, pp. 447-453, 2010.

[88]W. Jia, R. Hu, J. Gui, and Y. Lei, "Newborn footprint recognition using band-limited phase-only correlation", in *Proc. ICMB*, pp. 83-93, 2010.

[89]J.P. Campbell, "Speaker recognition: a tutorial", *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1437-1462, 1997.

[90]H. Hollien, *Forensic Voice Identification*. Academic Press, 2002.

[91]M.S. Nixon, T.N. Tan, and R. Chellappa, *Human Identification Based on Gait*. Springer, 2006.

[92]R. Plamondona and G. Loretteb, "Automatic signature verification and writer identification — the state of the art", *Pattern Recognit.*, vol. 22, no. 2, pp. 107-131, 1989.

[93]C.N. Liu, N.M. Herbst, and N.J. Anthony, "Automatic signature verification: system description and field test results", *IEEE Trans. Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 35-38, 1979.

[94]NIST Report on the United States Congress, Summary of NIST Standards for Biometric Accuracy, Temper, Resistance, and Interoperability, 2001.

[95]Face recognition access control system in the Olympic success of applications, http://news.frbiz.com/face_recognition_access_control-381706.html.

[96]L. Zhang, L. Zhang, and D. Zhang, "Finger-knuckle-print: a new biometric identifier", in *Proc. ICIP*, pp. 1981-1984, 2009.

[97]L. Zhang, L. Zhang, and D. Zhang, "Finger-knuckle-print verification based on band-limited

phase-only correlation", in *Proc. Int. Conf. Computer Analysis of Images and Patterns*, pp. 141-148, 2009.

[98] L. Zhang, L. Zhang, D. Zhang, and H. Zhu, "Online finger-knuckle-print verification for personal authentication", *Pattern Recognit.*, vol. 43, no. 7, pp. 2560-2571, 2010.

[99] L. Zhang, L. Zhang, D. Zhang, and H. Zhu, "Ensemble of local and global information for finger-knuckle-print recognition", *Pattern Recognit.*, vol. 44, no. 9, pp. 1990-1998, 2011.

[100] L. Zhang, L. Zhang, and D. Zhang, "MonogenicCode: a novel fast feature coding algorithm with applications to finger-knuckle-print recognition", in *Proc. Int. Workshop on Emerging Techniques and Challenges for Hand-Based Biometrics*, 2010.

[101] L. Zhang, L. Zhang, and X. Mou, "RFSIM: a feature based image quality assessment metric using Riesz transforms", in *Proc. ICIP*, pp. 321-324, 2010.

[102] L. Zhang, L. Zhang, Z. Guo, and D. Zhang, "Monogenic-LBP: a new approach for rotation invariant texture classification", in *Proc. ICIP*, pp. 2677-2680, 2010.

[103] M. Yang, L. Zhang, L. Zhang, and D. Zhang, "Monogenic binary pattern (MBP): a novel feature extraction and representation model for face recognition", in *Proc. ICPR*, pp. 2680-2683, 2010.

[104] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: a feature similarity index for image quality assessment", *IEEE Trans. Image Process.*, accepted.

[105] E. Stein and G. Weiss, *Introduction to Fourier Analysis on Euclidean Spaces*. Princeton, NJ: Princeton Univ. Press, 1971.

[106] C.D. Kuglin and D.C. Hines, "The phase correlation image alignment method", in *Proc. Int. Conf. Cybernetics and Society*, pp. 163-165, 1975.

[107] D. Gabor, "Theory of communication", *J. Inst. Elec. Eng.*, vol. 93, no. III, pp. 429-457, 1946.

[108] D. J. Field, "Relations between the statistics of natural images and the response properties of cortical cells", *J. Opt. Soc. Am. A*, vol. 4, no. 12, pp. 2379-2394, 1987.

[109] PolyU Finger-Knuckle-Print Database. http://www.comp.polyu.edu.hk/~biometrics/FKP.htm.

[110] N. Pinto, J.J. DiCarlo, and D.D. Cox, "How far can you get with a modern face recognition test set using only simple features?", in *Proc. CVPR*, pp. 2591-2598, 2009.

[111] N. Pinto, D.D. Cox, B. Corda, and D. Doukhan, "Why is real-world object recognition hard?: establishing honest benchmarks and baselines for object recognition", in *Proc. COSYNE*, 2008.

[112] A. Kong, "An evaluation of Gabor orientation as a feature for face recognition", in *Proc. ICPR*, pp. 1-4, 2008.

[113] T.S. Lee, "Image representation using 2D Gabor wavelet", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 10, pp. 957-971, 1996.

[114] M. Felsberg and G. Sommer, "The monogenic signal", *IEEE Trans. Signal Process.*, vol. 49, no. 12, pp. 3136-3144, 2001.

[115] M. Felsberg and G. Sommer, "The monogenic scale-space: a unifying approach to phase-based image processing in scale space", *J. Math. Imaging and Vision*, vol. 21, no. 1, pp. 5-26, 2004.

[116] M. Felsberg R. Duits, and L. Florack, "The monogenic scale space on a rectangular domain and its features", *Int. J. Computer Vision*, vol. 64, no. 2-3, pp. 187-201, 2005.

[117] M. Felsberg, "Optical flow estimation from monogenic phase", in *Proc. IWCM*, pp. 1-13, 2007.

[118] D. Zang and G. Sommer, "Signal modeling for two-dimensional image structures", *J. Visual Communication and Image Representation*, vol. 18, no. 1, pp. 81-99, 2007.

[119] L. Wietzke, G. Sommer, C. Schmaltz, and J. Weickert, "Differential geometry of monogenic signal representations", in *Proc. 2nd Int. Conf. Robot vision*, pp. 454-465, 2008.

[120] L. Wietzke and G. Sommer, "The 2D analytic signal", Technical Report, Kiel University, 2008.

[121] L. Wietzke, O. Fleischmann, and G. Sommer, "2D image analysis by generalized Hilbert transforms in conformal space", in *Proc. ECCV*, pp. 638-649, 2008.

[122] L. Wietzke, G. Sommer, O. Fleischmann, and C. Schmaltz, "The conformal monogenic signal of image sequences", in *Statistical and Geometrical Approaches to Visual Motion Analysis*, D. Cremers, et al., Eds., pp. 305-322, Springer, 2009.

[123] L. Wietzke, G. Sommer, and O. Fleischmann, "The geometry of 2D image signals", in *Proc. CVPR*, pp. 1690-1697, 2009.

[124] L. Wietzke and G. Sommer, "The signal multi-vector", *J. Math. Imaging and Vision*, vol. 37, no. 2, pp. 132-150, 2010.

[125] M. Mellor and M. Brady, "Phase mutual information as a similarity measure for registration", *Medical Image Analysis*, vol. 9, no. 4, pp. 330-343, 2005.

[126] K. Takaya, "Feature point correspondence of stereo images by monogenic phase", in *Proc. IEEE Pacific Rim Conf. Communications, Computers and Signal Processing*, pp. 272-275, 2007.

[127] J.A. Guerrero, J.L. Marroquin, and M. Rivera, "Adaptive monogenic filtering and normalization of ESPI fringe patterns", *Optics Letters*, vol. 30, no. 22, pp. 3018-3020, 2005.

[128] M. Unser, D. Sage, and D.V.D. Ville, "Multiresolution monogenic signal analysis using the Riesz-Laplace wavelet transform", *IEEE Trans. Image Process.*, vol. 18, no. 11, pp.

2402-2418, 2009.

[129] S. Held, M. Storath, P. Massopust, and B. Forster, "Steerable wavelet frames based on the Riesz transform", *IEEE Trans. Image Process.*, vol. 19, no. 3, pp. 653-667, 2010.

[130] M. Unser and D.V.D. Ville, "Wavelet steerability and the higher-order Riesz transform", *IEEE Trans. Image Process.*, vol. 19, no. 3, pp. 636-652, 2010.

[131] V. Sierra-Vazquez and I. Serrano-Pedraza, "Application of Riesz transforms to the isotropic AM-PM decomposition of geometrical-optical illusion images", *J. Opt. Soc. Am. A*, vol. 27, no. 4, pp. 781-796, 2010.

[132] S.L. Hahn, *Hilbert Transforms in Signal Processing*. Artech House, Boston, 1996.

[133] L. Moura and P. Monteiro, "Design method for FIR-based Hilbert transform filters suitable for broadband AM-SSB", *Electronics Letters*, vol. 38, no. 12, pp. 605-606, 2002.

[134] M. Abramowitz and I.A. Stegun, editors, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables.* New York: Dover, 1972.

[135] G. Krieger and C. Zetzsche, "Nonlinear image operators for the evaluation of local intrinsic dimensionality", *IEEE Trans. Image Process.*, vol. 5, no. 6, pp. 1026-1042, 1996.

[136] A.V. Oppenheim and J.S. Lim, "The importance of phase in signals", *IEEE Proc.*, vol. 69, no. 5, pp. 529-541, 1981.

[137] E.D. Castro and C. Morandi, "Registration of translated and rotated images using finite Fourier transforms", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 5, pp. 700-703, 1987.

[138] Q. Chen, M. Defrise, and F. Deconinck, "Symmetric phase-only matched filtering of Fourier-Mellin transforms for image registration and recognition", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 12, pp. 1156-1168, 1994.

[139] B. Reddy and B. Chatterji, "An FFT-based technique for translation, rotation, and scale-invariant image registration", *IEEE Trans. Image Process.*, vol. 5, no. 8, pp. 1266-1271, 1996.

[140] H. Foroosh, J.B. Zerubia, and M. Berthod, "Extension of phase correlation to subpixel recognition", *IEEE Trans. Image Process.*, vol. 11, no. 3, pp. 188-200, 2002.

[141] K. Takita, T. Aoki, Y. Sasaki, T. Higuchi, and K. Kobayashi, "High-accuracy subpixel image registration based on phase-only correlation", *IEICE Trans. Fundamentals*, vol. E86-A, no. 8, pp.1925-1934, 2003.

[142] K. Takita, M.B. Muquit, T. Aoki, and T. Higuchi, "A sub-pixel correspondence search technique for computer vision applications", *IEICE Trans. Fundamentals*, vol. E87-A, no. 8, pp. 1913-1923, 2004.

[143] H. Liu, B. Guo, and Z. Feng, "Pseudo-log-polar Fourier transform for image registration",

IEEE Signal Processing Letters, vol. 13, no. 1, pp. 17-20, 2006.

[144] W. Pan, K. Qin, and Y. Chen, "An adaptive-multilayer fractional Fourier transform approach for image registration", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 3, pp. 400-413, 2009.

[145] R. Bracewell, *The Fourier Transform and Its Applications*. McGraw-Hill, 1965.

[146] S. Venkatesh and R. Owens, "On the classification of image features", *Pattern Recognit. Letters*, vol. 11, no. 5, pp. 339-349, 1990.

[147] M.C. Morrone, J. Ross, D.C. Burr, and R.A Owens, "Mach bands are phase dependent", *Nature*, vol. 324, no. 6049, pp. 250-253, 1986.

[148] M.C. Morrone and R.A Owens, "Feature detection from local energy", *Pattern Recognit. Letters*, vol. 6, no. 5, pp. 303-313, 1987.

[149] M.C. Morrone and D.C. Burr, "Feature detection in human vision: a phase-dependent energy model", *Proc. R. Soc. Lond. B*, vol. 235, no. 1280, pp. 221-245, 1988.

[150] R. Owens, S. Venkatesh, and J. Ross, "Edge detection is a projection", *Pattern Recognit. Letters*, vol. 9, no. 4, pp. 233-244, 1989.

[151] P. Kovesi, "Image features from phase congruency", *Videre: J. Comp. Vis. Res.*, vol. 1, no. 3, pp. 1-26, 1999.

[152] P. Kovesi, "Phase congruency: a low-level image invariant", *Psychological Research Psychologische Forschung*, vol. 64, no. 2, pp. 136-148, 2000.

[153] P. Kovesi, "Invariant measures of image features from phase information", PhD Thesis, University of Western Australia, 1996.

[154] P. Kovesi, "Edges are not just steps", in *Proc.* ACCV, pp. 822-827, 2002.

[155] Z. Liu and R. Laganière, "Phase congruence measurement for image similarity assessment", *Pattern Recognit. Letters*, vol. 28, no. 1, pp. 166-172, 2007.

[156] L. Henriksson, A. Hyvärinen, and S. Vanni, "Representation of cross-frequency spatial phase relationships in human visual cortex", *J. Neuroscience*, vol. 29, no. 45, pp. 14342-14351, 2009.

[157] A.I. Zayed, *Handbook of Function and Generalized Function Transformations*. CRC Press, 1996.

[158] R. Snelick, U. Uludag, A. Mink, M. Indovina, and A.K. Jain, "Large-scale evaluation of multimodal biometric authentication using state-of-the-art systems", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 3, pp. 450-455, 2005.

[159] PolyU Palmprint Database. http://www.comp.polyu.edu.hk/~biometrics.

[160] Texas Instruments, *OMAP3530/25 Applications Processor*. 2009.

[161] DevKit8000 OMAP3530 Evaluation Kit,

http://www.embedinfo.com/english/product/devkit8000.asp.

[162] S.Pavlov and P. Belevsky, *Windows Embedded CE 6.0 Fundamentals*. Microsoft Press, 2008.

[163] S. Phung, *Professional Microsoft Windows Embedded CE 6.0*. Wiley Publishing, Inc., 2009.

[164] Texas Instruments, *Codec Engine Application Developer User's Guide*. 2007.

[165] Texas Instruments, *Codec Engine Algorithm Creator User's Guide*. 2007.

[166] Texas Instruments, *Codec Engine Server Integrator User's Guide*. 2007.

[167] Texas Instruments, *TMS320C64+ DSP Little-Endian DSP Library Programmer's Reference*. 2008.

[168] Texas Instruments, *TMS320C64+ Image/Video Processing Library*. 2003.

[169] Fingerprint Verification Competition (FVC2006), http://bias.csr.unibo.it/fvc2006, 2006.

[170] S. Yan, D. Xu, and X. Tang, "Face verification with balanced thresholds", *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 262-268, 2007.