THE HONG KONG
POLYTECHNIC UNIVERSITY

A FRAMEWORK FOR HIGH QUALITY PALMPRINT ACQUISITION AND EFFECTIVE PALM LINE EXTRACTION

SUBMITTED TO THE DEPARTMENT OF COMPUTING OF
THE HONG KONG POLYTECHNIC UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF PHILOSOPHY

by

WONG Ming Keung, Michael

January 2004

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 which has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____

WONG Ming Keung, Michael

ABSTRACT

Personal identification and verification both play a critical role in our society. Biometric technology identifies individuals by their physical or behavioral characteristics. One especially good biometric identifier is the human palmprint, which is rich in features such as principal lines, wrinkles and ridges. As yet, however, no device has been devised that is suitable for the real-time acquisition of palmprint images. This study has two major objectives: the design and implementation of a high quality palmprint acquisition device, and the proposition of a knowledge-based approach to palm line extraction.

The proposed palmprint acquisition system is designed for various civilian applications such as access control and for automatic teller machines. Its core components comprise a light source, lens, CCD sensor board, and a video frame grabber. With a resolution of 150 dpi, the system can obtain clear palm line features including principal lines, wrinkles and ridges. The most suitable light source and optimal system size were determined experimentally while a special–platform called the flat platen surface was built to serve as the user interface which guides the placement of palms during the acquisition process. Finally, we developed an online palmprint identification system which uses the proposed system for the palmprint acquisition. That system now controls access to our own laboratory.

The second objective of our research was the establishment of a novel palm line extraction approach based on knowledge of palm line structures. The palm line extraction

method starts with the adaptive thresholding technique to obtain a preliminary line map of a palmprint image. Next, we defined the properties of line structures and line segments. Then, we designed the searching strategies to exploit the structural information of a palm. Some rules were established for solving problems such as isolated points and broken lines. Finally, the major palm lines can be extracted effectively using the above steps. For the palm line matching, we first define a bounding box to limit the searching area. Next, we divide each line into shorter lines called nodes and perform angle comparisons node by node to get the similarity scores. We introduced the idea of point shifting and node shifting in order to take care of the shifting and translation problems. In addition, we employed a bidirectional matching scheme to further enhance the results. The experimental results show that our palm line extraction framework is effective. We think that palm line extraction can contribute to the classification of palmprints, or could even be used on some new applications such as automatic palmistry (fortune telling).

ACKNOWLEDGEMENTS

I would like to give my sincere thanks to my supervisor, Prof. David Zhang whose supervision and guidance kept my research on the right track and whose comments and criticisms ensured my research had every chance of success.

In addition, I would like to express my deep gratitude to Prof. Zhaoqi Bian at Tsinghua University for his useful advice.

I would also like to thank Wenxin Li, Adams Kong, Bo Pang, Xiangqian Wu, and Guangming Lu, at the *Biometric Research Centre* of *The Hong Kong Polytechnic University* and the *Biocomputing Research Lab* of *Harbin Institute of Technology*. They provided me with many resources including journal papers and websites which helped me complete this research. Their seminars discussing on various biometrics technologies broadened my research horizons.

Also I am grateful to everyone who participated in the Biometrics Data Collection, and gave their palmprints for our research work on this project.

This thesis is dedicated to my wife Camille, whom I love dearly, and who is a constant source of inspiration. Finally, I would like to express my greatest gratitude to my family for their care and support.

TABLE OF CONTENTS

PART I: AN OVERVIEW

PART II: PALMPRINT ACQUISITION DEVICE

LIST OF FIGURES

Part I: An Overview

1 INTRODUCTION

1.1 The Need for Biometrics

Personal identification and verification both play a critical role in our society. Today more and more business activities and work practices are computerized. eCommerce applications such as eBanking or security applications such as building entrance demand fast, real time, accurate personal identification. Traditional knowledge-based or token-based personal identification or verification are tedious, time-consuming, inefficient and expensive.

Knowledge-based approaches use "something that you know" (such as passwords and personal identification numbers [1]) for personal identification; token-based approaches, on the other hand, use "something that you have" (such as passports or credit cards) for the same purpose. Tokens (e.g. credit cards) are time consuming and expensive to replace. Passwords (e.g. for computer logon and email account) are hard to remember and keep track of. According Gartner [2], a company spends about US$14 to US$28 to handle a password reset and about 19% of help desk calls are related to the password reset problem. This may suggest that the traditional knowledge based password protection is unsatisfactory. Thorough discussions on the weaknesses of the password authentication methods are mentioned in [3]. Since these approaches are not based on any inherent attribute of an individual in the identification process, they are unable to differentiate between an authorized person and an impostor who fraudulently acquires the

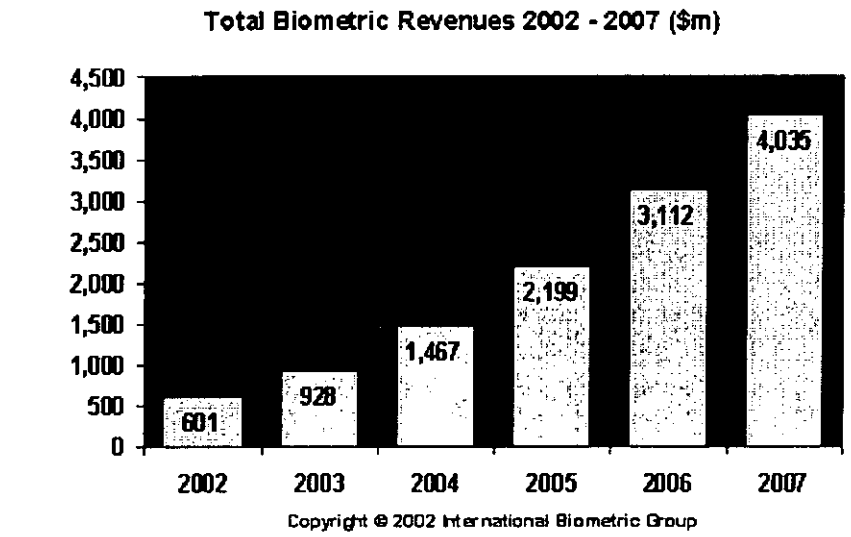
"token" or "knowledge" of the authorized person. These shortcomings gave way to biometrics identification or verification systems to become the focus of the research community in the recent years: biometric systems based on iris, hand geometry and fingerprint [4-8] were developed in the past decade.

Biometrics involves the automatic identification of an individual based on his physiological or behavioural characteristics. The first commercial system, Identimat, was developed in 1970s, as part of a time clock at Shearson Hamill, a Wall Street investment firm [9] and measured the shape of the hand and the lengths of the fingers. At the same time, fingerprint-based automatic checking systems were widely used in law enforcement by the FBI and US government departments. Advances in the hardware such as faster processing power and greater memory capacity made biometrics more viable. Since the 1990s, iris, retina, face, voice, palmprint, signature and DNA technologies have joined the biometric family [1, 10].

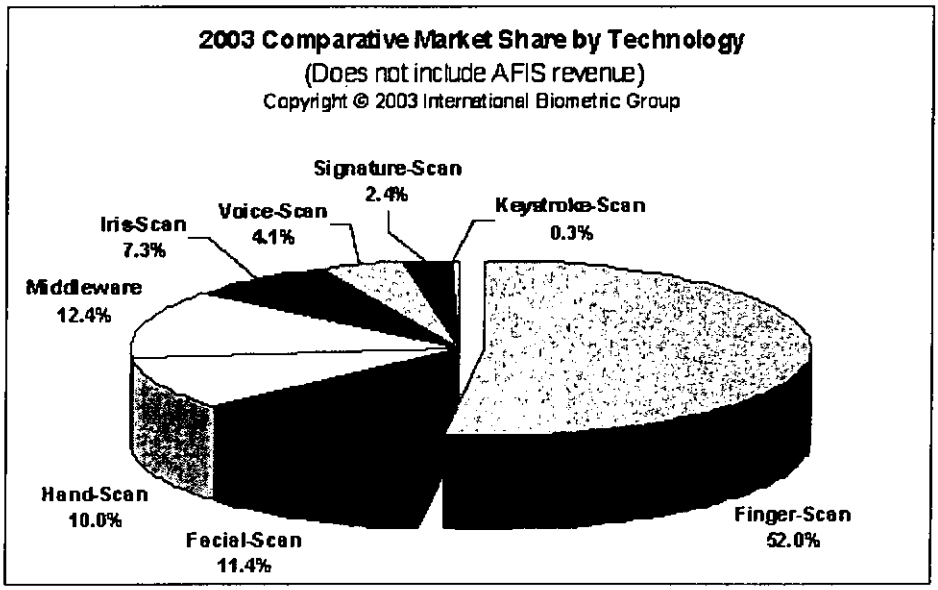
After the 911 terrorist attacks, the interest on biometrics-based security solutions and applications has increased dramatically, especially in the need to spot potential criminals in crowds. This further pushes the demand on the development of different biometrics products. For example, some airlines have implemented the iris recognition technology in airplane control rooms to prevent any entry by unauthorized persons. In 2004, all Australian international airports will implement passport using face recognition technology for airline crews and it will eventually becomes available to all Australian passport holders [11].

Fig. 1.1 (a) shows predicted total revenues from biometric for 2002-2007. A steady rise in revenues is predicted, from US $928 million in 2003 to US $4,035 million in 2007. Fig. 1.1 (b) shows Comparative Market Share by different biometric technologies for the year 2003. With a 2003 market share of 52% [12], fingerprint technology is the world's

most widespread biometric technologies.



(a)



(b)

Fig. 1.1 (a) Total biometric revenues prediction in 2002-2007, (b) Comparative market share by biometric technologies in 2003. (*Source: International Biometric Group*).

1.2 Biometric System Architecture

A biometric system is essentially a pattern recognition system which makes a personal identification by determining the authenticity of a specific physiological or behavioral characteristic possessed by the user [13]. Normally, personal characteristics such as fingerprints, palmprints or 3-D hand geometry are obtained through a sensor and fed into the pattern recognition engine to return a result of success or failure. Fig. 1.2 shows the architecture of a typical biometric system. In general, every biometric system consists of the following four stages: 1) Signal acquisition 2) Signal preprocessing 3) Feature extraction and 4) Feature matching.

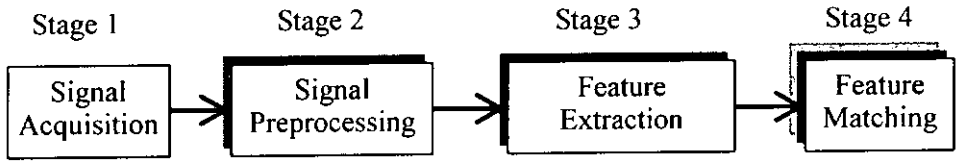


Fig. 1.2 Four stages of a biometric system.

1) Signal acquisition – biometric data is obtained from an input device. The quality of signals is very important since it forms the raw input for subsequent processing.
2) Signal preprocessing – enhancement of the signals is performed in this stage, including segmentation, noise reduction, and rotation and translation normalization.
3) Feature extraction – the features defined possess the stable and unique properties of low intra-class difference and high inter-class difference. These features are used to create a master template which is stored in the system database.
4) Feature matching – a matching score is obtained by matching the identification template against the master templates. If the score is less than a given threshold, the user is authenticated.

1.2.1 Operation Mode of a Biometric System

A biometric system is usually operated in three modes: enrollment, identification and verification. But some systems only have either identification or verification modes.

Enrollment – Before a user can be verified or identified by the system, he/she must be enrolled by the biometric system. The user's biometric data is captured, preprocessed and feature extracted as shown in stages 1 to 3 of Fig. 1.2. Then, the user's template is stored in a database or file system.

Identification – This refers to the identification of a user based solely on his/her biometric information, without any prior knowledge about the identity of a user. Sometimes it is referred to 1-to-many matching, or recognition. It will go through stages 1 to 3 to create an identification template. Then the system will retrieve all the templates from the database for the feature matching in stage 4. A result of success or failure is given finally. Generally, accuracy decreases as the size of the database grows.

Verification – This requires that an identity (ID card, smart card or ID number) is claimed, and then a matching of the verification template with master templates is performed to verify the person's identity claim. Sometimes verification is referred to a 1-to-1 matching, or authentication.

1.2.2 Performance Evaluation of a Biometric System

Performance evaluation of a biometric system usually refers to False Acceptance Rate (FAR), False Rejection Rate (FRR) and Equal Error Rate (EER) [14]. FAR refers to a situation where a non-registered user gains access to a biometrically protected system, while FRR refers to a situation where a registered user fails to gain rightful access to a biometrically protected system at the first attempt. EER is represented by a percentage in

which the FAR and FRR are equal. EER provides a unique measurement which fairly compares the performance of different biometric systems. In an ideal system, there are no false rejections and no false acceptances but as yet no such system has been developed. In general, a biometric system can be operated at different levels of security: high, medium or low. Fig. 1.3 illustrates a hypothetical Receiver Operating Characteristics (ROC) curve and typical operating points for different biometric applications [15]. The ROC curve of a system illustrates the false reject rate (FRR) and the false acceptance rate (FAR) of a matcher at all operating points (threshold, T). Each point on an ROC curve defines the FRR and FAR operating at a particular threshold. In fact, the security threshold depends mainly on the application context of a biometric system. If it is designed for a high security environment such as the entrance of a nuclear station, then more false rejections with less false acceptances occur. The ideal case is to operate the system in both low FRR and low FAR.

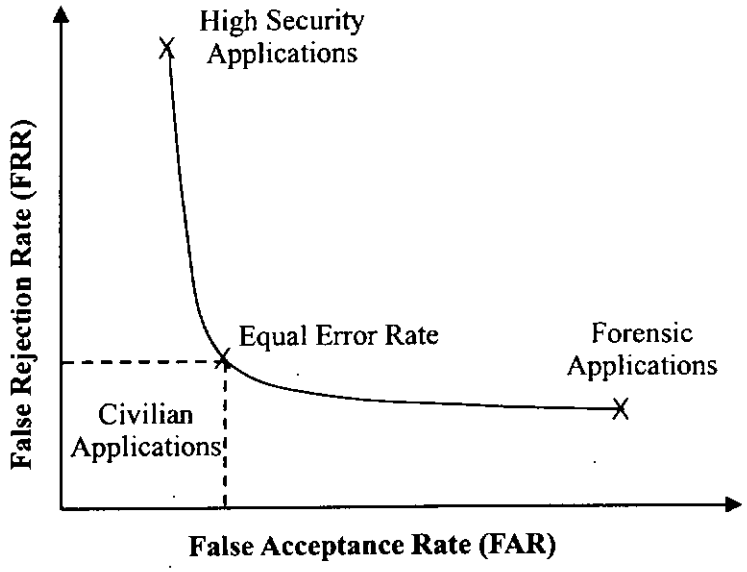


Fig. 1.3 A hypothetical ROC curve and typical operating points for different biometric applications.

1.2.3 Perception on Biometric Identifier

Seven factors affect the determination of a biometric identifier, including: universality, uniqueness, permanence, collectability, performance, acceptability, and circumvention. Table 1.1 summarises how three biometrics experts perceive five common biometric technologies [16].

Table 1.1 Perception of five common biometric technologies by three biometrics experts (*Source: An Introduction to Biometric Recognition, IEEE* ©).

	Face	**Fingerprint**	**Hand Geometry**	**Iris**	**Palmprint**
Universality	High	Medium	Medium	High	Medium
Uniqueness	Low	High	Medium	High	High
Permanence	Medium	High	Medium	High	High
Collectability	High	Medium	High	Medium	Medium
Performance	Low	High	Medium	High	High
Acceptability	High	Medium	Medium	Low	Medium
Circumvention	High	Medium	Medium	High	Medium

The applicability of a specific biometric technique depends heavily on the application domain. These factors can influence the selection of a biometric system.

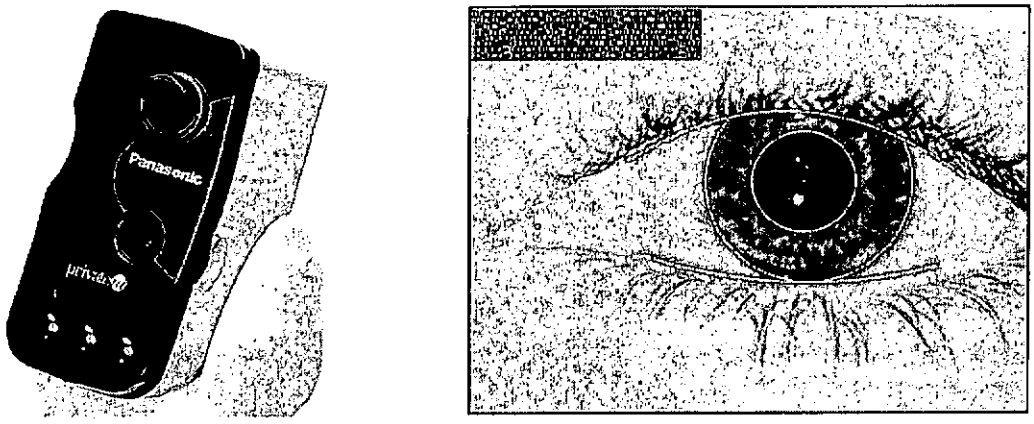
- *Universality* – each person should have the characteristic.
- *Uniqueness* – no two persons should be the same in terms of the characteristic.
- *Permanence* – characteristic should neither change in a long period of time.
- *Collectability* – the characteristic can be measured quantitatively.
- *Performance* – accuracy and speed of identification, and robustness of the system.
- *Acceptability* – refers to the extent to which people are willing to accept a particular biometric identifier in their life.
- *Circumvention* – refers to how easy it is to fool the system.

1.3 Different Biometric Technologies

A significant limitation on existing biometric-based personal identification systems is their imperfect performance in terms of accuracy and acceptance. These systems sometime falsely accept an impostor (FAR) and falsely reject a genuine user (FRR) [1]. An ideal biometric system that satisfies all the criteria of high accuracy, high user acceptance and low cost, is yet to be developed. Each existing system has its own strengths and limitations. There is no definite answer for which biometric system is the most appropriate one as the selection of biometrics is an application dependent decision. The following shows different types of biometric technology and products available on the market.

1.3.1 Iris

Iris recognition is the most effective biometric technology, being able to accurately identify the identities of more than thousand persons in real-time [13, 17]. The iris is the colored ring that surrounds the pupil. A camera using visible and infrared light scans the iris and creates a 512-byte biometric template based on characteristics of the iris tissue, such as rings, furrows, and freckles.



(a) (b)

Fig. 1.4 (a) Iris acquisition device (*Source: Iridian Technologies, Inc.*), (b) an iris image with its *IrisCode* and eye-localization graphics (*Source: J. Daugman*).

An iris recognition system such as IrisAccess™ from Iridian Technologies, Inc. [18] provides a very high level of accuracy and security. Its scalability and fast processing power fulfils the strict requirements of today's marketplace but it is expensive and users regard it as intrusive. It is suitable for high security areas such as nuclear plants or airplane control rooms. On the other hand, it is not appropriate in areas which require frequent authentication processes, such as logging onto a computer. Fig. 1.4 (a) shows an iris camera manufactured by Panasonic®, while Fig. 1.4 (b) shows an iris with its *IrisCode* and eye-localization graphics [19].

1.3.2 Face

Compared to others biometrics, face verification is low cost, needing only a camera mounted in a suitable position such as the entrance of a physical access control area. For verification purposes it captures the physical characteristics such as the upper outlines of the eye sockets, the areas around the cheekbones, and the sides of the mouth. Face-scanning is suitable in environments where screening and surveillance are required with minimal interference with passengers.

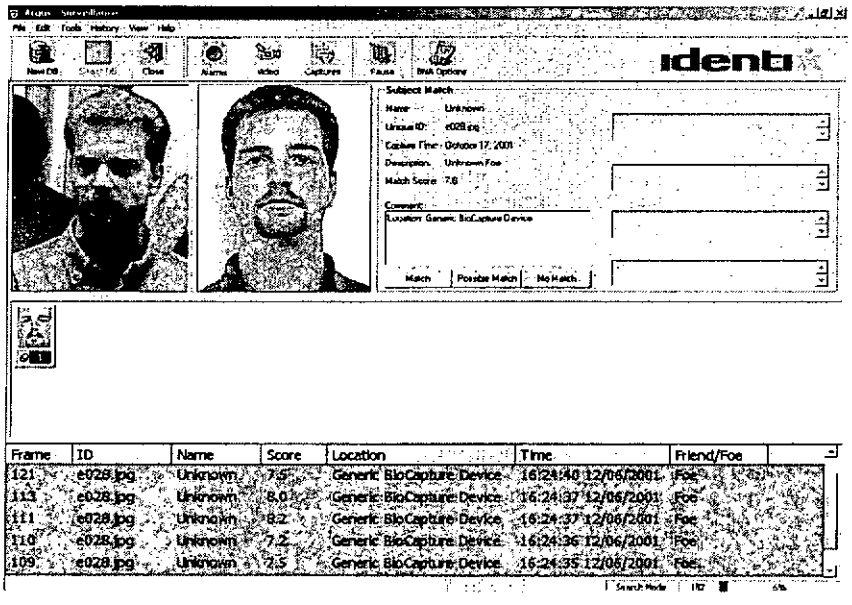


Fig. 1.5 A screen shot of the Face Recognition Software. (*Source: Identix Inc.*)

The State of Virginia in the U.S. has installed face-recognition cameras on Virginia's beaches to automatically record and compare their faces with images of suspected criminals and runaways [20]. However, the user acceptance of facial scanning is lower than that of fingerprints, according to a IBG Report [21]. Fig. 1.5 shows a screen shot of the face recognition software called FaceIt® ARGUS. It is an off-the-shelf facial recognition system that detects and identifies human faces as they pass through a camera's field of view, which can help the detection of suspect from a database.

1.3.3 Fingerprint

The commercial application of biometric devices began in the early 1970s when a system called Identimat, which measured the shape of the hand and length of the fingers, was used as part of a time clock at Shearson Hamill, a Wall Street investment firm [9]. Subsequently, hundreds of Identimat devices were used to establish identity for physical access at secure facilities run by different governmental offices such as U.S. Naval Intelligence and the Department of Energy.

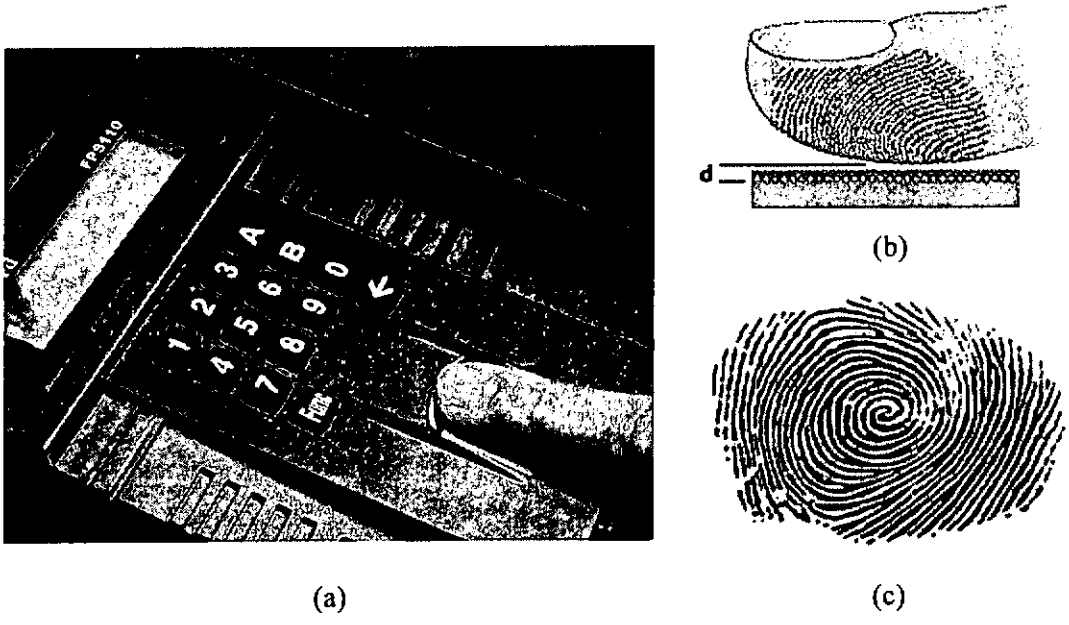


Fig. 1.6 (a) a fingerprint sensor, (b) a cross section diagram (*Source: Veridicom Inc.*), and (c) a typical fingerprint image.

Human fingerprint is made up of ridges that take the shape of loops, arches, and whorls. Minutiae are the points on a fingerprint where a ridge ends or splits into two. The most promising minutiae points are extracted from an image to create a template, usually between 250 to 1,000 bytes in size [8, 15, 22-26]. It is the most widely used biometric technology in the world, has a market share of 52% in 2003 [12]. Its small chip size, ease of acquisition and high accuracy make it the most popular biometrics technology since the 1980s. However, some people may have fingerprints worn away due to hand work and some old people may have many small creases around their fingerprints, lowering the system's performance. In addition, the fingerprint acquisition process is sometimes associated with criminality, causing some users to feel uncomfortable with it.

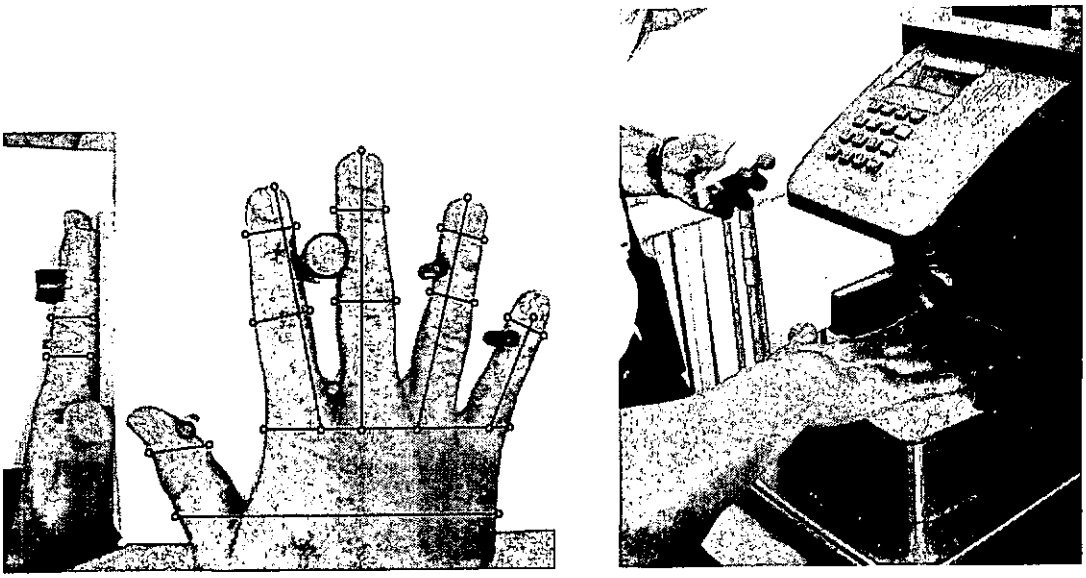
A typical capacitive based fingerprint sensor is shown in Fig. 1.6 (a), where (b) illustrates how it is touched by a finger tip, and (c) exhibits a typical fingerprint image. Popular applications including the time attendance and employee management systems and physical access controls install in the entrance of buildings.

1.3.4 Hand Geometry

Hand geometry requires only small feature size, including the length, width, thickness and surface area of the hand or fingers of a user, as shown in Fig. 1.7 (a) [6].

There is a project called INSPASS (Immigration and Naturalization Service Passenger Accelerated Service System) [27] which allows frequent travelers to use 3D hand geometry at several international airports such as Los Angeles, Washington, and New York. Qualified passengers enroll in the service to receive a magnetic stripe card with their hand features encoded. Then they can simply swipe their card, place their hand on the interface panel, and proceed to the customs gate to avoid the long airport queues. Several housing construction companies in Hong Kong have adopted the Hand Geometry

for the employee attendance record in their construction sites, as shown in Fig. 1.7 (b). A smart card is used to store the hand shape information and employee details. Employees verify their identities by their hand features against the features stored in the smart card as they enter or exit the construction site. This measure supports control of access to sites and aids in wage calculations.



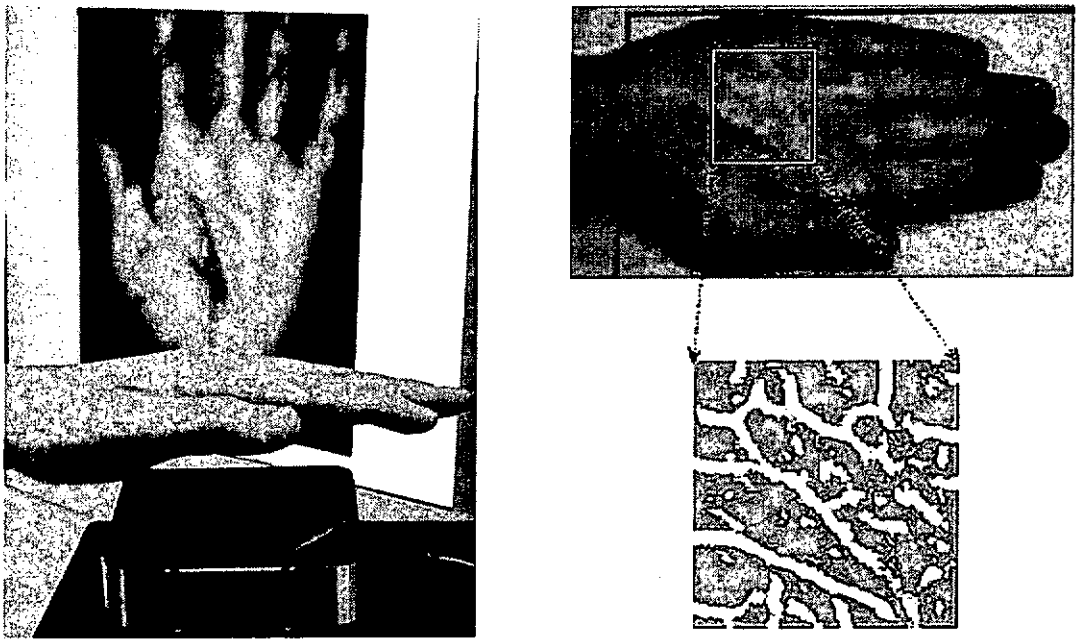
(a) (b)

Fig. 1.7 (a) the features of a hand geometry system (*Source: A. K. Jain*), (b) A hand geometry system operating in a construction site in Hong Kong.

Hand geometry has several advantages over other biometrics, including small feature size, and low cost of computation as a result of using low-resolution images [6, 28]. In spite of its current widespread use (10% market share in 2003 [12]), the current hand geometry system suffers from high cost and low accuracy [13]. In addition, uniqueness of the hand features is not guaranteed, making it unfavorable to be used in one-to-many identification applications.

1.3.5 Palm Vein Patterns

In August 2002 Fujitsu Laboratories Limited announced a new type of biometric authentication technology which verifies a person's identity by the pattern of the veins in his/her palms [29]. They claim that, except for their size, palm vein patterns are personally unique, and do not vary over the course of a person's lifetime. They use infrared light to get the image of a palm when the hand is held over the sensor device. By testing 1400 palm profiles collected from 700 people, the system achieved a false rejection rate of 1% and a false acceptance rate of 0.5% with an equal error rate of 0.8% [30]. The use of this technology would enable convenient biometric authentication for different types of applications such as log-in verification for access to sales, technical or personal data. Fig. 1.8 (a) shows the Palm Vein Pattern Biometric Authentication System while Fig. 1.8 (b) shows a sample vein patterns. Since it is a new biometric technology, there is not much confidence from the public to use it.



(a) (b)

Fig. 1.8 Palm vein pattern technology: (a) Palm Vein Pattern Biometric Authentication System, (b) sample vein patterns. (*Source: Fujitsu Laboratories Limited*)

1.4 Why Palmprint?

Every biometric technology has its merits and limitations and no one is the best for every application domain. Nonetheless, as this Section outlines, palmprint does have advantages over other hand-based biometric technologies.

1.4.1 Definition of Palmprint

A palm is defined as the inner surface of a hand between the wrist and the root of fingers. A palm is rich in features: principal lines, winkles, ridges, singular points and minutiae points, as shown in Fig. 1.9, make up the palmprint. It is very much larger than a finger tip but is covered with the same kind of skin.

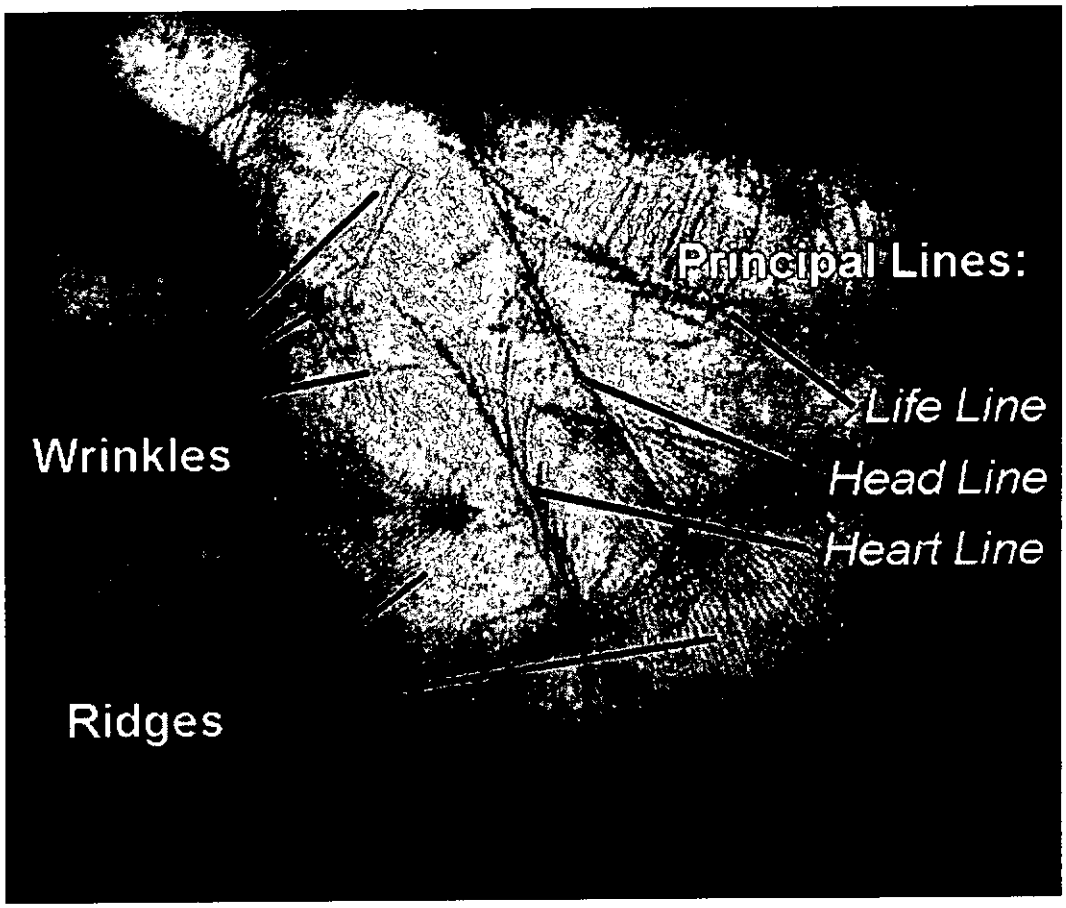


Fig. 1.9 Three levels of line features from a palm

1.4.2 Palmprint Advantages

Palmprints have advantages over other hand-based biometric technologies:

(a) The palm provides a larger surface area compared with the fingerprint so that more features can be extracted.

(b) An individual is less likely to damage a palm than a fingerprint and the line features of a palm are resistant to throughout one's lifetime.

(c) Small amounts of dirt or grease sometimes appear on an individual's finger, adversely affecting the performance of fingerprint verification. This problem does not arise in the extraction of palmprint features since the highest resolution for the proposed system is only 150 dpi.

(d) Compared with 3D hand geometry, palmprint has more unique features that can be used for personal identification, so a better performance can be expected.

Many unique features of a palmprint image can be used for personal identification including principal lines, wrinkles, ridges, minutiae points, singular points and texture, in which all of them are useful in palmprint representation [31]. Palmprints also use a much lower resolution imaging sensor (150 dpi) compared with a fingerprint (500 dpi), with the result that computation is much faster in both the preprocessing and feature extraction stages. In view of these advantages, we proposed to use palmprint features for personal identification in different real-time applications.

1.5 Organization of the Thesis

This thesis is organized in four parts. Part I has two chapters, providing an overview of biometric technology and underscoring the advantages of palmprint research. It first introduces different biometric technology and defines the palmprint identification system. Then it sketches the history on the development of the palmprint acquisition device and

investigates our proposed framework. Part II has three chapters, outlining the design and requirement analysis on the palmprint acquisition device. Next, the internal components, user interfaces and the appearance design are investigated. Then it evaluates the performance of the palmprint acquisition device. Part III has four chapters, reporting a novel recognition engine based on palm lines extraction. It first introduces some existing palmprint preprocessing techniques, then it discusses the proposed idea on the palmprint signal preprocessing, which involve the binarization and edge detection on the palmprint image. We define rules and strategies from the information obtained from the palm lines' structures, for the line features extraction. Then we propose the matching strategies and report the experimental results from the recognition engine. Lastly, Part IV has one chapter that offers our conclusions and suggests directions for further research.

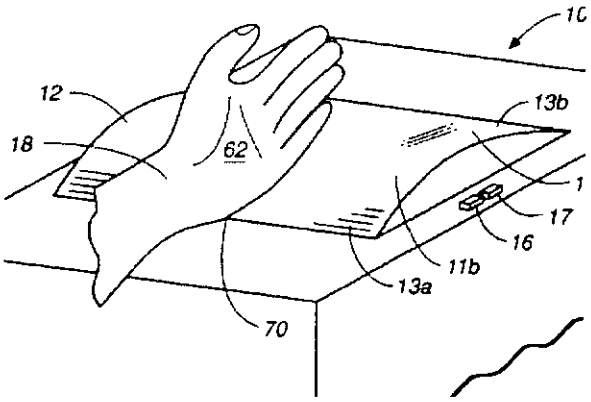
2 RESEARCH ON PALMPRINT TECHNOLOGY

2.1 Palmprint Research Workgroups

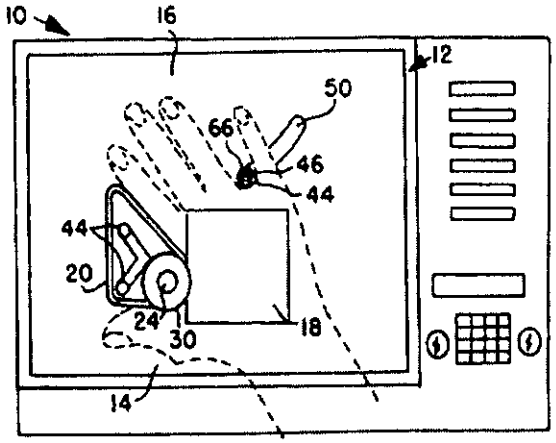
Although the palmprint is a new member of the biometric family, there are a variety of workgroups and organizations interested in the palmprint research, including hardware design, algorithm development and forensic applications. Indeed, there has been a patent for a palm positioning mechanism for more than 30 years.

2.1.1 Palmprint Patents

There are patents granted in the past few decades about different design on the palm or palmprint acquisition and identification apparatus [32-36]. Some of them are concerned with the design of the apparatus (palmprint or palm lines acquisition device) using an optical imaging mechanism; some of them have designed a special controlling tool tailored for the palm shape measurements such as palm positioning and system actuating mechanisms (Fig. 2.1). It seems however that none of those patented designs is available on the market as a commercial product for the biometric identification or verification applications.



(a)

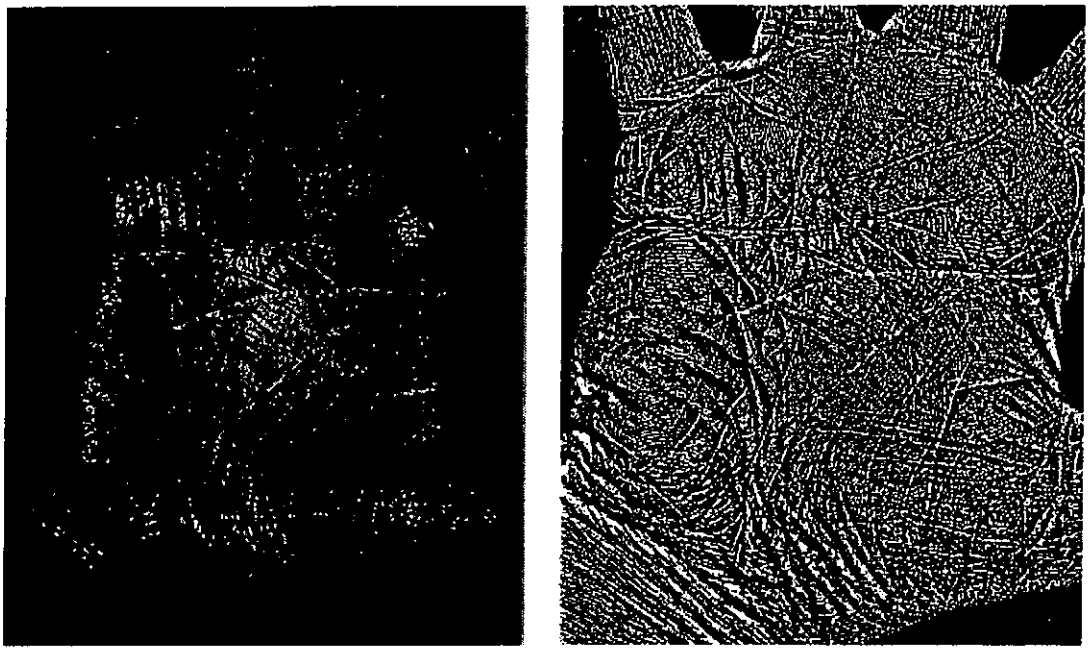


(b)

Fig. 2.1 (a) an Electro-optic palm scanner (*Source: US Patent No. 5528355*), (b) a palm positioning and actuating mechanism (*Source: US Patent No. 4357597*).

2.1.2 Commercial Sectors

Several companies have developed inkless palmprint identification / verification systems for criminal applications [37-41]. Their systems use live scanners to acquire high resolution images from which many detailed features, such as ridges, singular points and minutiae points, can be extracted and matched with the features from latent prints. Building on their previous design, Biometric Partners Inc. [42] has developed a touchless palm sensor which has a recognition distance of up to 300 mm, while maintaining high contrast image resolution and real-time imaging (Fig. 2.2).



(a) (b)

Fig. 2.2 Palm image acquired by a live scanner: (a) raw image, (b) binarization of the raw image. (*Source: Biometric Partners Inc.*)

2.1.3 Research Teams

Several research teams are investigating palmprints [43-50] focusing on the recognition methods of inked and/or inkless palmprints. One of these research teams [45] uses a digital scanner to obtain inkless palmprint images, obviating the problem of an inked palmprint but the technique is not appropriate for real-time applications because the scanning process is time-consuming. The fact that there is no suitable palmprint acquisition system available on the market for civil applications has motivated us to design a palmprint acquisition system for palmprint research.

2.2 Acquisition of Palmprint Data

2.2.1 Offline Method

Before the development of online capture device, palmprint data was obtained by offline method, as shown in Fig. 2.3. Offline method involved collecting samples by ink in a user's palm and pressing it onto a sheet of white paper. After the ink has dried, the palmprint image on the paper is digitized by a scanner and stored in the personal computer [51]. This method is not, however, suitable for real time application such as physical access control and the quality of the palmprint image is not satisfactory.

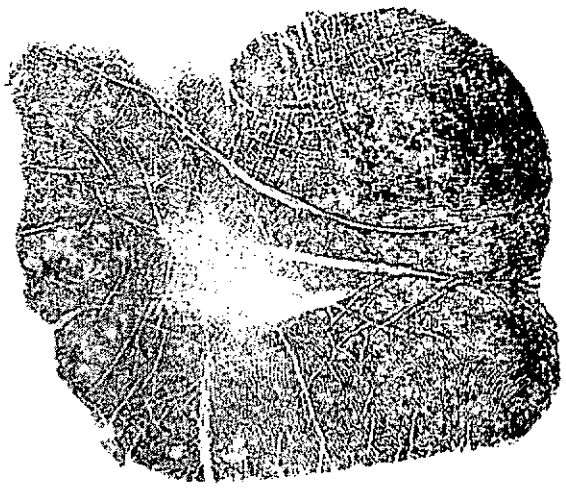


Fig. 2.3 An inked palmprint image from offline acquisition method.

2.2.2 Online Method

In December 1999, led by Prof. Zhang at the Department of Computing in The Hong Kong Polytechnic University, Niu et al. [10] started to design their first online palmprint capture device. It was the first attempt to make an online device for palmprint research. The device is made by a plastic box, a light source, a mirror, a glass plate and a CCD camera, as shown in Fig. 2.4. After repeated testing, however, they found that the image formed through the mirror is not as good as a direct reflection because the second surfaced mirror creates a ghost image. In addition, the glass plate used to hold the palm

distorts the surface of the skin of the palm so that palm lines are not good enough for the feature extraction. A better device is needed in order to get a better image quality for further processing.

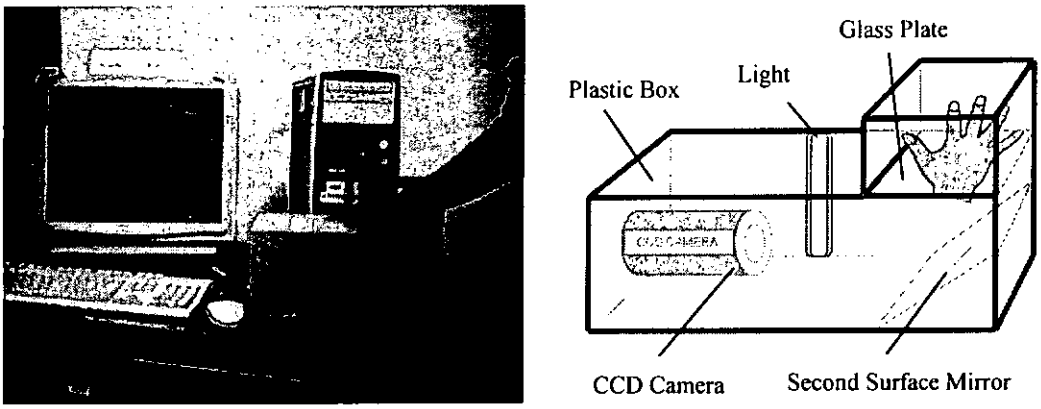


Fig. 2.4 First online palmprint acquisition device: (a) appearance, (b) architecture.

2.3 Proposed Framework

Like most biometric systems, palmprint identification system is a four-stage process involving signal acquisition, signal preprocessing, feature extraction, and feature matching. Functionally, we can place the components in two groups: 1) hardware and peripherals, and 2) software. So our framework is designed in two layers of abstraction: Layer 1 is the *Palmprint Acquisition Device* composed of a user interface and sensor, while layer 2 is a *Line Based Recognition Engine* which carries out image preprocessing, feature extraction and matching, as shown in Fig. 2.5. A further breakdown of the framework is shown in Fig. 2.6. Layer 1 is discussed in Part II of this thesis while layer 2 is described in Part III.

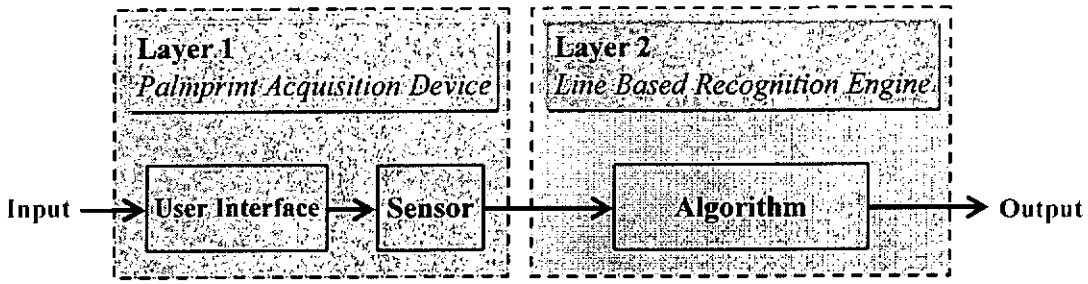


Fig. 2.5 Overview of the proposed framework

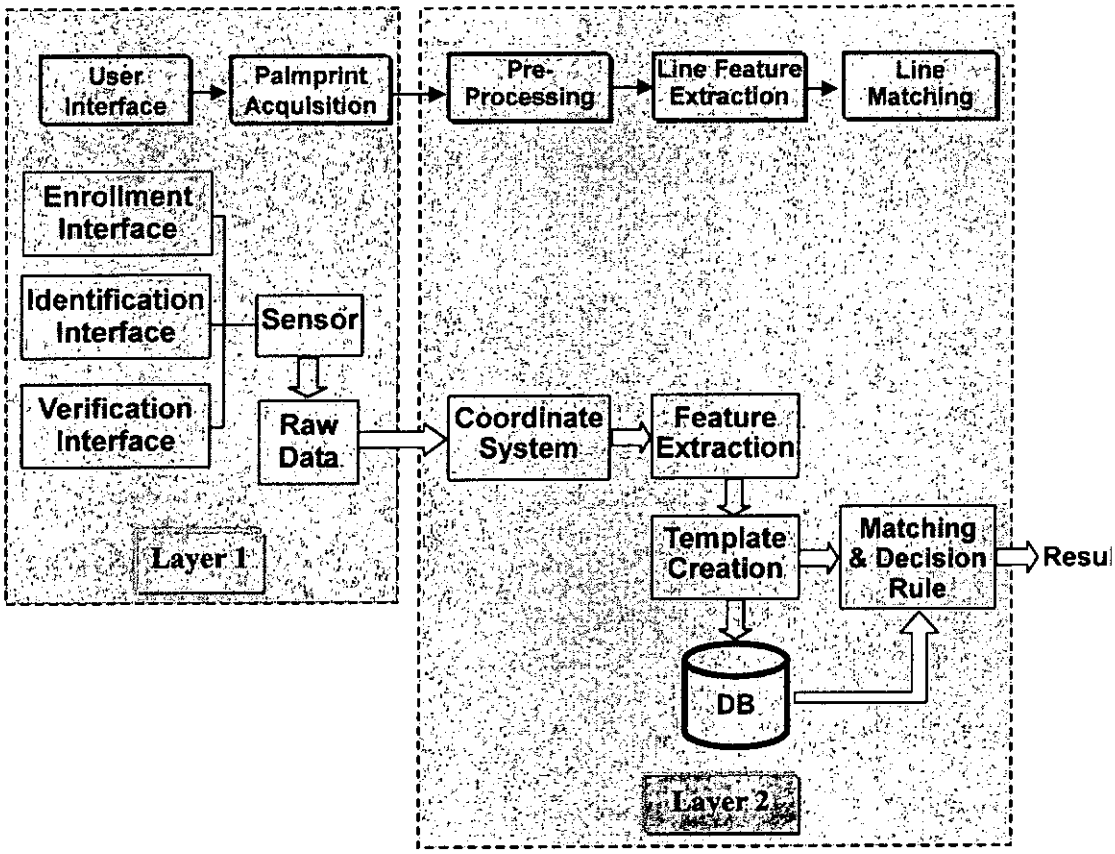


Fig. 2.6 Breakdown of the proposed framework.

2.3.1 Layer 1 – Palmprint Acquisition Device

The main objective of Layer 1 is to design a high quality palmprint acquisition device with a comfortable user interface. For the user interface, there is a specially designed flat platen surface for holding the palm in alignment. It acts as the input channel for the user and system, for the acquisition of the palmprint signal from the user interface. The sensor

unit consists of a light source, a lens, a CCD sensor board, and a video frame grabber. The light from the object (palm) passing through the aperture is converged by the lens to form an image. A CCD sensor transforms the photon signal into an electrical signal. A video frame grabber processes the analog signal from the CCD sensor and an Analog to Digital Conversion (ADC) take place in the on-board chip. The proposed framework uses an Intel Celeron 950 MHz CPU, 128 MB RAM and storage of 32 MB Disk on Module.

2.3.2 Layer 2 –Line Based Recognition Engine

The main objective of Layer 2 is to design a line based recognition engine that based on the palm lines' structural information. It consists of stages of: 1) palmprint signal preprocessing, 2) line feature extraction, and 3) line matching. In the palmprint signal preprocessing stage, noise is reduced using image enhancement procedures and then a coordination system is established to locate the palmprint data from the raw image. Finally, we binarize the image to get the preliminary line map of a palmprint image. In the line feature extraction stage, major lines are extracted to create master templates for purposes of identification or verification. A knowledge based approach is introduced which uses the structural information of a palmprint image to enhance the performance of palm line extraction. In the line matching stage, the identification template is compared with the master templates and the system determines whether they have come from the same person.

2.4 Possible Applications Using Palmprint

Biometrics can be used in both customer and employee oriented applications such as ATMs, airports, and time attendance management with the goals of improving the workflows and eliminating fraud. We propose the use of a palmprint identification system to supplement or even replace existing services and methods such as smartcard or

password-based authentication. Our proposed system is most appropriate for the following applications: (i) audit trail and anti-fraud activities, (ii) transportation system, and (iii) access control and human resource management.

2.4.1 Audit Trail and Anti-fraud

Retail Banking

The internal operation of retail banking such as daily authentication process can be replaced by using biometric technology. Some banks have implemented an authorization mechanism for different hierarchy of staff by swiping their badge for audit trail purpose. But a supervisor's badge may be stolen, loaned to other members of staff or even lost. Our proposed system eliminates these kinds of problems by placing a palmprint identification device on each supervisors' desk. When a junior member of staff has a request, it is transmitted to the supervisors' computer for palmprint approval and automatically recording.

ATM

Although the existing authentication environments for ATM operations are imperfect, the password based authentication method is the most effective and widely used mechanism. ATM are customer oriented applications where palmprint can replace the existing password based authentication methods.

2.4.2 Transportation

Airports

As mentioned in Chapter 1 Section 1.3.4, INSPASS allows frequent travelers to use the 3D hand geometry at several international airports such as Los Angeles to speed up the boarding procedures [27]. Palmprint recognition systems could do the same.

2.4.3 Access Control and Human Resource Management

Building Access Control

The purpose of building access control is to monitor and verify the identities of individuals who enter or leave restricted areas. These kinds of systems can be customer or employee-oriented. Our system can complement or replace the use of keys or badges. It can keep track of who has entered to particular areas because palmprints cannot be shared or borrowed.

Time and Attendance Management

"Buddy punching" is the workplace custom in which employees punch time cards for friends who may be late or absent from work. It is a problem for many large companies which blame it for the loss of hundreds of millions of dollars every year. The use of a palmprint identification system to replace time cards could eliminate this type of fraud.

2.4.4 Other Applications

Citizen ID Program

It is a trend for governments to use biometric technology on the issuance of citizen identity cards. In Hong Kong, a project called Smart Identity Card System (SMARTICS) uses the fingerprint as the authentication identifier [52]. Efficient government services using SMARTICS will provide increased security and faster processing times on different operations such as driver license or border crossing. We think that palmprint is effective to be used on similar applications.

Part II: Palmprint Acquisition Device

3 REQUIREMENT ANALYSIS ON THE ACQUISITION DEVICE

To develop a real-time, on-line palmprint identification system requires a palmprint scanner that can quickly capture high quality palmprint images, which is a fundamental step of palmprint research. However, limited research effort has been put into designing and implementing palmprint acquisition system. The proposed palmprint acquisition system is designed for personal identification based on palmprint recognition. In this Chapter, we analyze the requirements of a novel palmprint acquisition device. The structure of the proposed system is shown in Fig. 3.1. There is a user interface for the input of the palm. A set of optical components would then work together to obtain the signals from the palm. The analog signal is converted into a digital signal using an A/D converter and the digitized signals are stored in the main memory of our system.

There are specific criteria for our system: user interface requirements and optical system requirements.

User Interface Requirements

- *Easy and intuitive to user* – users should feel comfortable when they are using the system since the level of user acceptance is very important nowadays for any customer based applications.
- *Size* – the size is dependent on applications, but it should be as small as possible.

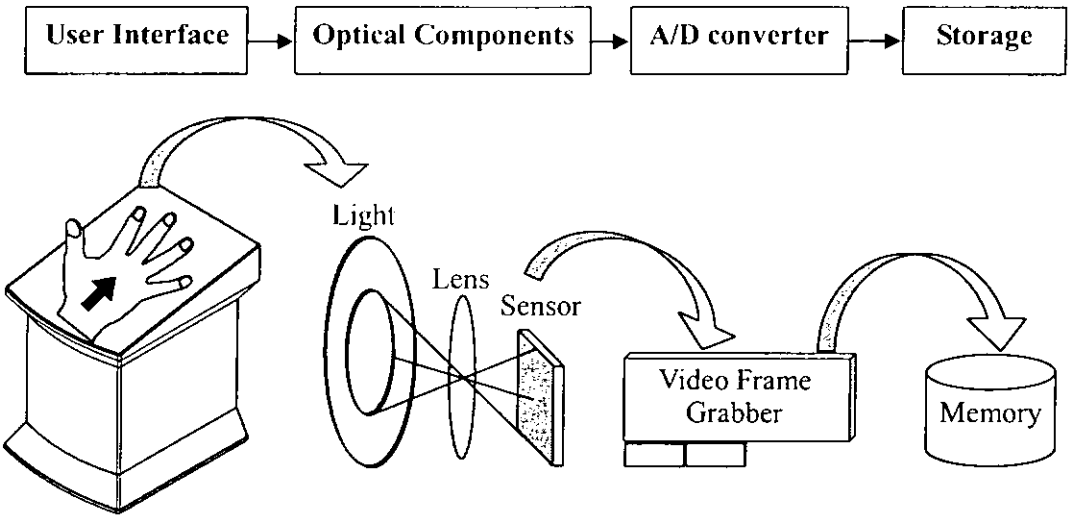


Fig. 3.1 The structure of the palmprint acquisition system.

Optical System Requirements

- *Image quality* – high quality images must be obtained such that the subsequent image processing for the recognition can be performed properly.
- *Real time processing* – the response time of the system should be fast enough to provide a development platform for civilian applications.

Having defined the criteria of our system, the remainder of this chapter will focus on how those requirements could be achieved in different levels on the system development.

3.1 Analysis on the User Interface of the Device

Acceptability to users is essential in the selection of a biometric identification technology. Factors affecting the acceptability of a system include user friendliness, ease of use, and a non-intrusive design. If users feel uneasy with a device, they are reluctant to cooperate with the whole identification system. As a result, the sensors might acquire some noise signals which would seriously affect the system performance.

In view of this circumstance, we need to pay special attention to the design of different components in the user interface, including:

1) A panel to hold a palm

   It is important to have a place to hold the palm so that the user feels good during the acquisition process. This panel should help the acquisition of stable signal for the pattern recognition engine in the subsequent stages.

2) Palm positioning guides

   Some simple instructions can be easily followed by the users without any difficulties in the whole acquisition process.

3.2 Analysis on the Size and the Appearance of the Device

The appearance of a device plays another important role on the selection of biometric system. We should determine an appropriate size and appearance for the user interface panel and, indeed, the entire device. The considerations include:

1) Appropriate size of the user interface panel

   The size of the user interface panel should allow different sized palms to be similarly placed. It should faithfully acquire sufficient features from palms of different sizes.

2) Size of the whole device

   The device must be of a size appropriate for use in different applications. Some applications such as building access control will allow a larger device while some indoor applications such as computer logon will require a smaller device.

3.3 Analysis on Possible Optical System

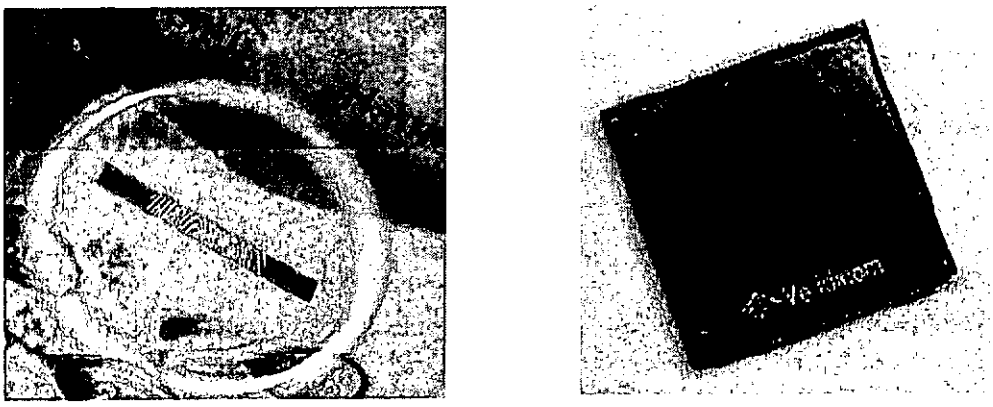
Every biometric system normally has a sensor for the acquisition of biometric signals. Palmprint, a 2-D signal, can be acquired with various imaging sensors now on the market. For the purpose of identifying an appropriate tool as the acquisition device in our framework, we only focus on the imaging sensors usually used in biometric systems, such as capacitive sensors, flatbed scanners, and CCD cameras.

3.3.1 Capacitive Sensor

Most fingerprint devices use capacitive sensors. These contact mode sensors that, depending on the manufacturer, require the user either to hold the finger still on the sensor surface or to swipe it across the sensor, allowing the user's finger tip skin surface data can be read, digitized and transferred to the device for further processing.

Shown in Fig. 3.2 (a) is the world's smallest capacitive based fingerprint sensor (*14 mm* (W) x *4.3 mm* (D) x *1.2 mm* (H)), developed by Fujitsu [53]. The user swipes a finger across a thin sensor strip and the fingerprint image is recorded in sequence row by row. Each metal electrode acts as one capacitor plate and the contacting finger acts as the second plate. A passivation layer on the surface of the device forms the dielectric between these two plates.

Another vendor, Veridicom [54], produced a sensor which has dimensions of *24 mm* (W) x *24 mm* (D) x *1.4 mm* (H). It is a type of touch sensor which acquires the fingerprint image by holding the user's hand stationary on the sensor surface during image acquisition, as shown in Fig. 3.2 (b). Both the Fujitsu and the Veridicom sensors provide 500 dpi resolution images.

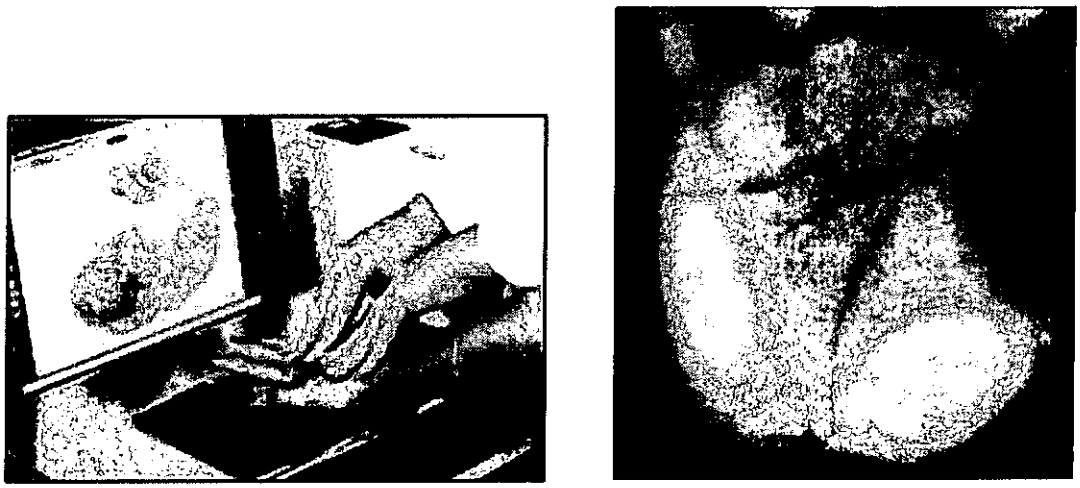


(a) (b)

Fig. 3.2 Fingerprint sensors: (a) capacitive based swipe sensor (*Source: Fujitsu*), and (b) capacitive based touch sensor (*Source: Veridicom*).

3.3.2 Scanner

A flatbed scanner typically operates by moving a scan head slowly across a target glass plate using a belt attached to a stepper motor. The resolution is dependent on the number of sensors in a single row detector array and the precision of the stepper motor [55]. Scanners capture high resolution images at a slow speed. Fig. 3.3 (a) shows a commercial palm scanner.



(a) (b)

Fig. 3.3 (a) A commercial palm scanner (*Source: Cross Match Technology Inc.*), (b) Palmprint images obtained with a general purposed scanner.

3.3.3 Charged Couple Device (CCD) Camera

A CCD sensor converts optical signals into electrical signals. These signals are then transported across the chip and stored in the sensor array where the charge is converted to voltage and amplified, and an analog-to-digital converter digitizes the signals [56]. CCD technology is mature and images obtained are distortion-free, low noise and high quality in terms of fidelity and light sensitivity while the acquisition speed of a CCD camera is the same as that of a camera.

The CCD camera is usually connected to a video frame grabber for the A/D conversion and the digitized data is stored in the main memory of a computer. The CCD camera and video frame grabber determine the actual speed of the acquisition process. A

popular video frame grabber can capture 25 frames per second (fps), i.e. 0.04*s*, which is adequate for capturing real-time image for personal identification systems.

3.3.4 Analysis of the Imaging Sensors

Fig. 3.3 (a) is an image obtained using Cross Match ID 1500™ palm scanner [37], while Fig. 3.3 (b) is a palmprint image obtained using a general purposed flatbed scanner in grayscale at a resolution of 150 dpi. It can be seen that there are deformed skin surfaces on the left and lower right parts of the palm while the central part is not clear enough for feature extraction. As the skin of the palm is not a flat surface, the image obtained using this scanner is not acceptable. Although we do not have such a large capacitive sensor to acquire palmprint image for comparison, it can be expected that the details from the central part of a palmprint may similarly be lost due to the skin structure of the palm is not a flat surface. In addition, the slow acquisition process of the scanner makes it undesirable for our proposed system.

Fig. 3.4 shows a palmprint image obtained using a CCD camera. It is a high quality image with uniform illumination. The palm does not need to touch the imaging sensor during the acquisition period. Owing to the weaknesses of the capacitive sensor and scanner sensor, we propose to use CCD camera as the acquisition core components of the palmprint acquisition device. Table 3.1 summarizes the advantages and disadvantages of the above sensors.

In fact, the contactless design of the palmprint acquisition interface plays an important role in obtaining high quality palmprint image which does not distort the image with uniform illumination. Owing to the weaknesses of the capacitive sensor and scanner sensor, we propose to use the approach of CCD camera as the core component for image acquisition.

Table 3.1 Comparison of different imaging sensors.

	Advantages	Disadvantages
Capacitive sensor	● readily available ● low price ● high resolution ● real time processing	◆ sensor area too small to fully cover the palm ◆ cannot capture features located in the central part of the palm
Flatbed scanner	● readily available ● low price ● usually very high quality (but depends on the CCD sensor used)	◆ slow speed in the whole acquisition process ◆ some of the skin is deformed ◆ central part of the palm is not clear
CCD camera	● readily available ● low price ● usually very high quality (but depends on the CCD sensor used) ● real time processing (25 fps) ● captures detailed palm lines and features without deformation	◆ need to implement the lighting environment ourselves

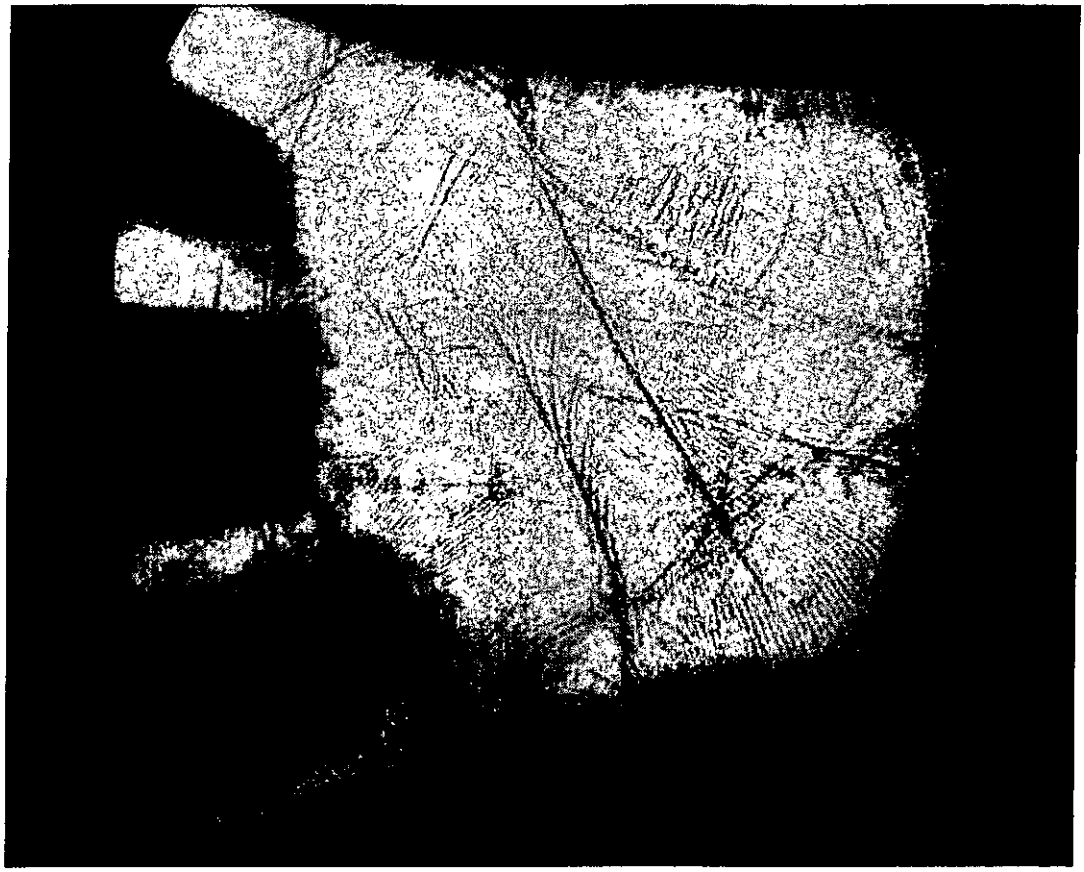


Fig. 3.4 Palmprint image obtained using a CCD Camera.

3.4 Analysis of Real Time Processing

The real time processing capability of the palmprint acquisition is controlled by the processing power of the CCD sensor chip, and the processing speed of the video frame grabber chip. For the CCD sensor, there is an electronic shutter which controls the integration time for each frame of an image at the fastest of 10 μs. For the video frame grabber, we digitize the image using the Fusion™ 878A video decoder [57]. A video frame grabber is an image digitizer which takes an analog signal as input and converts it into a digital signal. The main advantage of using a video frame grabber is its fast frame rate and short integration time [58-59]. The integration time of a CCD sensor is dependent on the aperture of the lens, focal length and lighting. By the use of CCD camera connected to a video frame grabber, the image can be obtained at 0.04s on a 25

frames per second (fps) video frame grabber. Since the CCD operates at $10\mu s$ while the video frame grabber can only operates at $0.04s$, the fastest speed of our acquisition device is $0.04s$. We believe that this speed is more than enough for a real-time processing system for a palmprint identification system.

There are 25 frames per second (fps) and 625 lines on each frame. For those 625 lines, there are only 576 lines used for the picture element. If the square pixel is used with 4:3 aspect ratio, the maximum pixels per line are:

576 x 4/3 = 768 pixels per line

That is, the maximum pixels in a frame for PAL are:

768 x 576 = 442368 pixels

A high speed data bus (PCI version 2.2 or PC104) is used to interface with the computer for the high data rate generated by the video frame grabber. The data rate requirement is:

1 frame = 768 x 576
= 442368 Bytes (1 Byte per pixel)
1 second = 25 fps x 768 x 576
= 11059200 Bytes

The palmprint acquisition device transmits 11.06 MBytes per second of data to the host machine. Since the PCI bus channel can operate at 132 MB/s, it can handle the data produced by the video frame grabber sufficiently. That is, the later stages of the palmprint system can take the stream of data from the main memory directly without a bottleneck.

3.5 The Proposed Acquisition Device: Working Principles

Wildes et. al. proposed a framework of image acquisition for an iris recognition system [60], as shown in Fig. 3.5. The diagram illustrates three subcomponents: physical capture, illumination and positioning. To guarantee a high image quality, they used a low light

level black and white silicon intensified target (SIT) camera that the signal obtained from the camera is then digitized by a frame grabber. For the lighting, they use an array of lights arranged around the camera which is pointing to the object's eye to be acquired. In order to provide a constant lighting environment, they use two filters for ensure uniform illumination of the eye (iris).

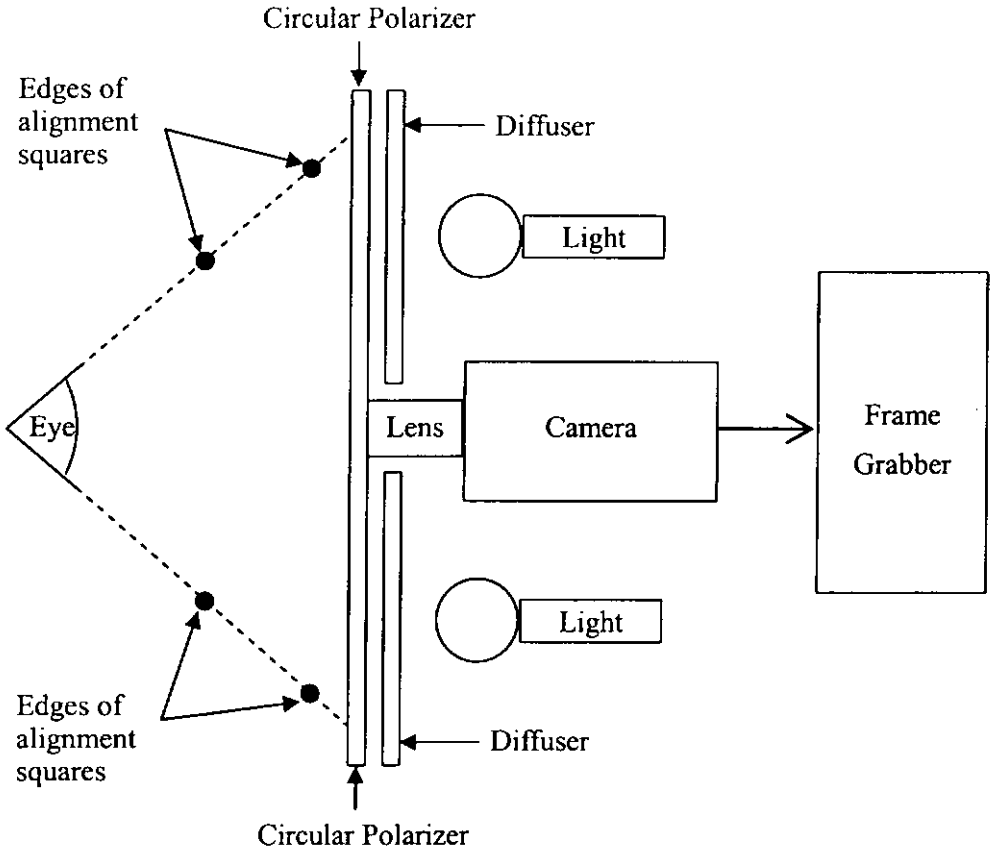


Fig. 3.5 The framework of image acquisition components for an iris recognition system.

The framework of the proposed palmprint acquisition system is similar to Wildes et. al. The core components of our proposed acquisition device are light source, a lens, a CCD sensor board, and a video frame grabber. Fig. 3.6 provides an overview of how the analog signal is obtained from the source (palm) through a lens to the CCD sensor, and then transformed to digital signal by the video frame grabber. Although all the above

components can be bought off-the-shelf, the components must be carefully selected and arranged for optimum performance.

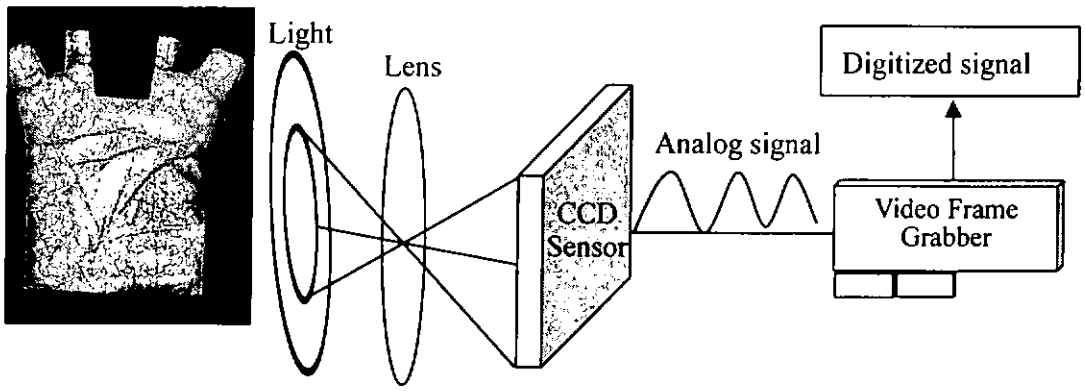


Fig. 3.6 The working principles of a CCD sensor with video frame grabber

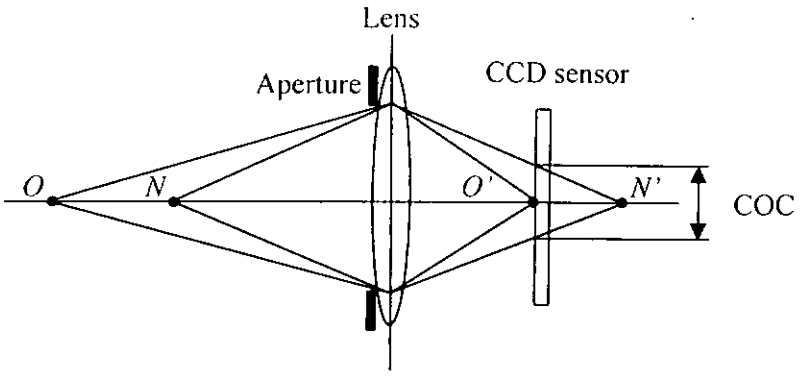
3.5.1 Light Source

Lighting is one of the most important factors in the quality of an image. The type of light source should illuminate the palm uniformly. If, for example, having one side of the palm is too bright while the other is too dim will not produce images suitable for extracting palmprint features.

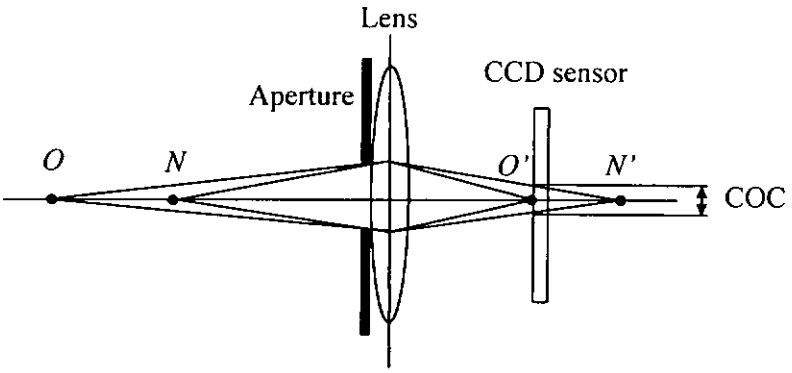
3.5.2 Lens

Another important factor in an optical system is the lens. In order to cover the whole CCD sensing area and to minimize distortion from the rim of lens, the lens format (diagonal length) should be larger than the format of the CCD sensor. In our proposed palmprint acquisition device we eliminate lens distortion, close by the rim, by using a 1/2" lens format with a 1/3" CCD sensor format.

Another important issue of a lens is the Depth of Field (DOF). Palms are not flat surfaces so the focus from the object plane spans a range of distances. This problem can be addressed by calculating the required DOF to allow a wider range of distances of the palm to be brought into focus.



(a)



(b)

Fig. 3.7 Large COC is formed by a large aperture in (a), where it can be reduced by using smaller size of aperture in (b).

When a CCD sensor is placed to capture a sharp image from an object (O), the near object (N) forms a blur spot called a circle of confusion (COC) [61], see Fig. 3.7 (a). O' is the image of O, and N' is the image formed by N. As shown in Fig. 3.7 (b), decreasing the size of the aperture can effectively decrease the blur spot, COC. An aperture is described by its F-stop (f_n) which is defined as:

$$f_n = efl / ca, \qquad (3\text{-}1)$$

where efl is the effective focal length and ca is the clear aperture of the lens. A smaller blur spot can be produced by using a smaller aperture, shorter focal length and longer object distance, but while setting a very small aperture does increase the DOF, the

reduction in light entering the CCD sensor also reduces contrast. We calculated an optimized aperture value to ensure that enough DOF and sufficient light are provided in the optical system to achieve the balanced system performance.

3.5.3 CCD Sensor Board

The purpose of a CCD sensor is to convert light signals into electrical signals. The picture sharpness and resolution of the CCD sensor is related to the Horizontal TV Lines (HTVL) which are determined by the pixel size and number of pixels in the CCD sensor. In order to obtain the highest image quality for palmprint feature extraction, one should choose a CCD sensor with a very high HTVL resolution.

3.5.4 Video Frame Grabber

A video frame grabber is an image digitizer which takes an analog signal from CCD sensor as input and converts it into a digital signal. The main advantage of using a video frame grabber is its fast frame rate and short integration time [57-58].

4 USER INTERFACE AND OUTLOOK DESIGN

This Chapter focuses on the design of user interface, and the appearance, specifically the size and shape, of the device.

4.1 User Interface

The acceptability to users of a particular biometric technology is important in the selection of a biometric system [1, 13]. Different factors which affect the acceptability of a device such as user friendliness, ease of use, and non-intrusive design. Our primary concern was to design a user interface that was intuitive and easy to use, and that would not stress users or make them feel uncomfortable. The user interface panel must also be of a design that allows it to be used in the same way by people with palms of different sizes. We applied the principle of iterative design and problem solving techniques on the development of the user interface so that the final design is the most suitable.

4.1.1 First Design – Rectangular Shape Hole

The first attempt on the design and implementation of an online palmprint capture device is led by Prof. Zhang et al. [10]. The user interface is a rectangular glass plate. It is designed without pegs to guide the placement of a palm. Users are free to put their hand on the surface of the glass plate during the acquisition process, as see Fig. 4.1 (a). However, this approach has similar problem to the flatbed scanner, i.e. may cause image

distortion, which is unsuitable for palmprint feature extraction. In view of this problem, we removed the glass plate, see Fig. 4.1 (b) and found that the image quality is much better. However, there are variations in the way individual users placed their hands on the device, image quality was still not stable with some users even placing their palms outside the image sensing area. This problem gave rise to our novel flat platen surface design.

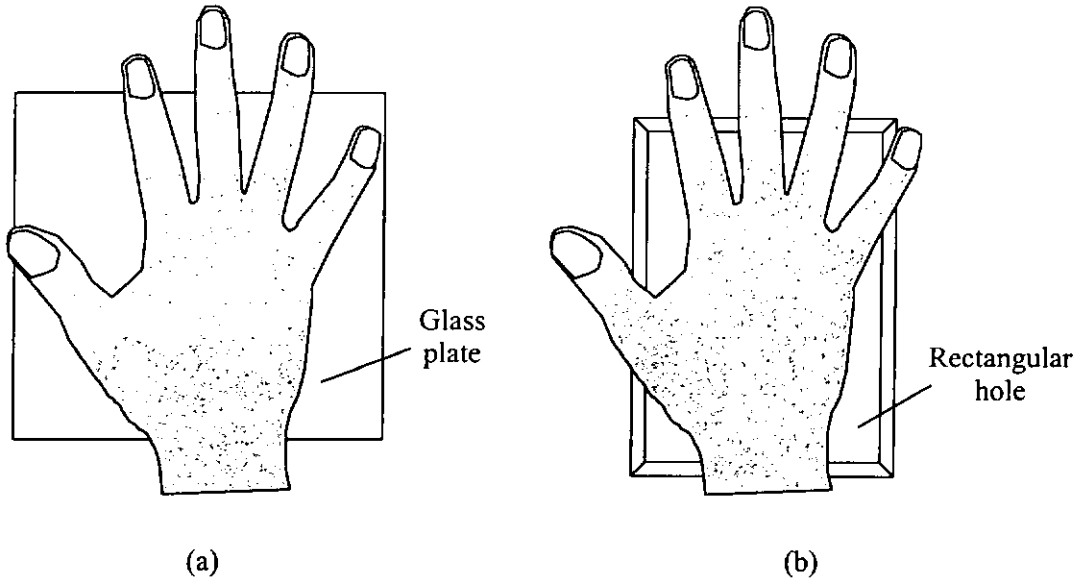


Fig. 4.1 A hand is put on the glass plate in (a), while (b) shows the removed glass plate.

4.1.2 Second Design – Flat Platen Surface

We designed an enhanced user interface called flat platen surface for palmprint acquisition. It is in appropriate size, comfortable to use and able to accommodate most people. In order to provide better guidelines for correctly placing a palm, we added pegs as shown in Fig. 4.2. The dotted line area is cropped from the flat platen surface to allow the CCD camera underneath to acquire the palmprint image. We carried out tests to determine a suitable size for the flat platen surface and positioning of the pegs. With the current design, we can obtain images from large or small hands, as shown in Figs. 4.3. As the images show, all the important line features are obtained.

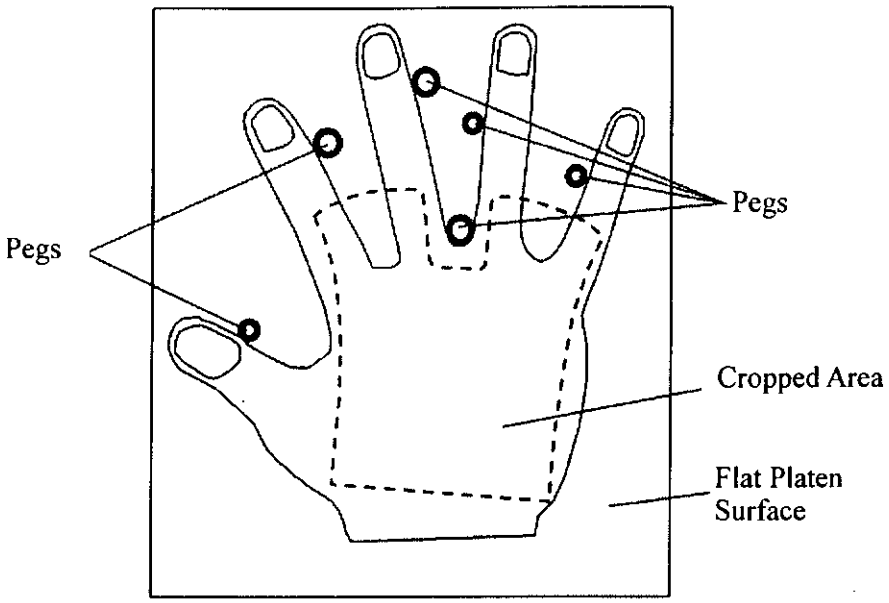


Fig. 4.2 Flat platen surface designed for the palm image acquisition with six pegs for palm guidance

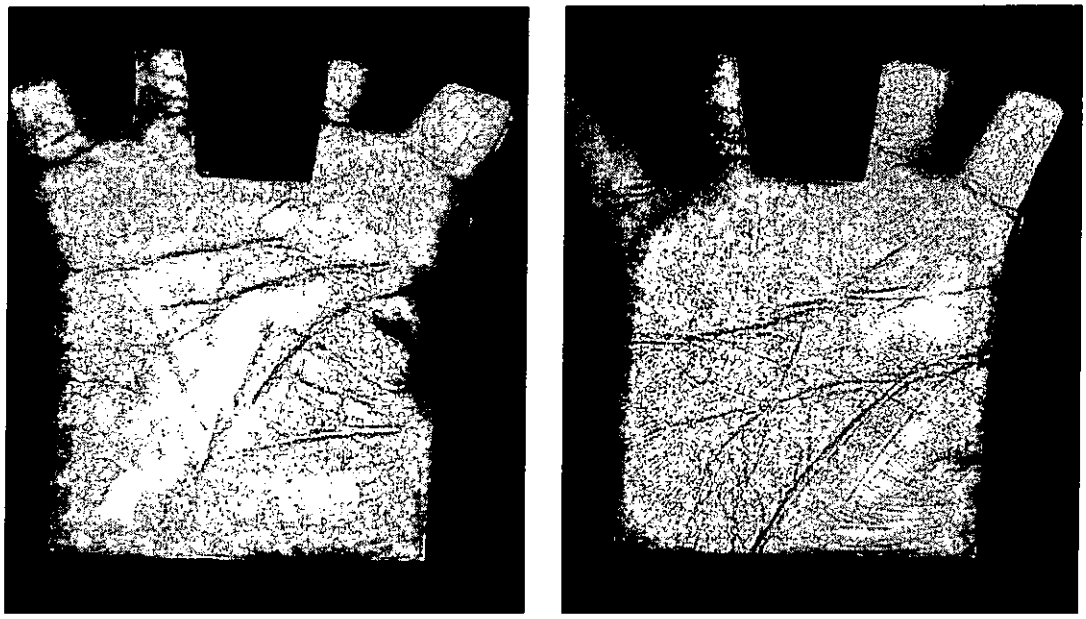


Fig. 4.3 (a) a small hand, (b) a large hand.

4.1.3 Final Design – Enhanced Flat Platen Surface

Our experience has shown that a proliferation of pegs on the flat platen surface can cause confusion among users as to how they should place their palms. The ultimate design takes account of this, and the use of pegs balances user convenience and system stability. As

shown in Fig. 4.4, the flat platen surface requires only three pegs: a large triangular peg to guide the middle and ring fingers; one round-shaped peg on the left to guide the index finger, and one peg on right to guide the little finger. We provide a clear and easy-to-use user interface and a stable palmprint alignment positioning mechanism. The design of the flat platen surface is thus satisfactory from both the user and system perspectives.

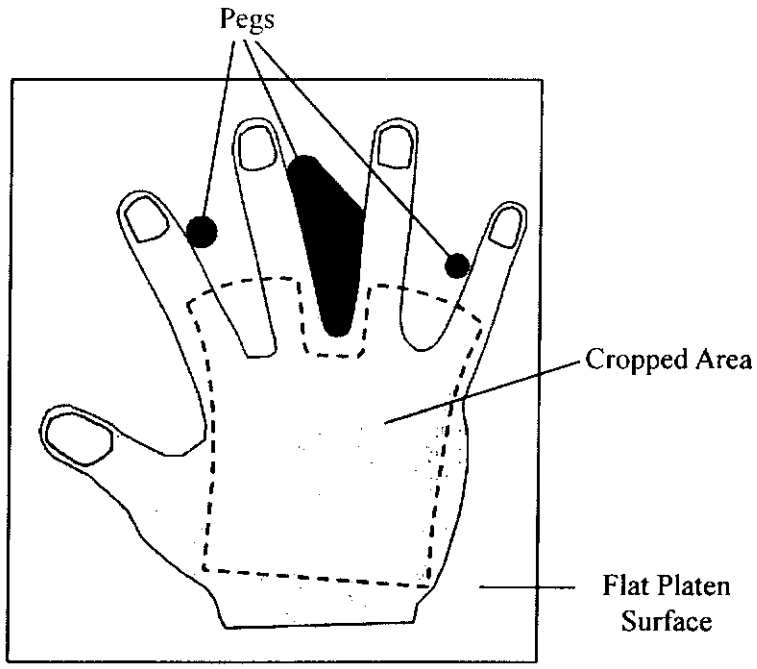


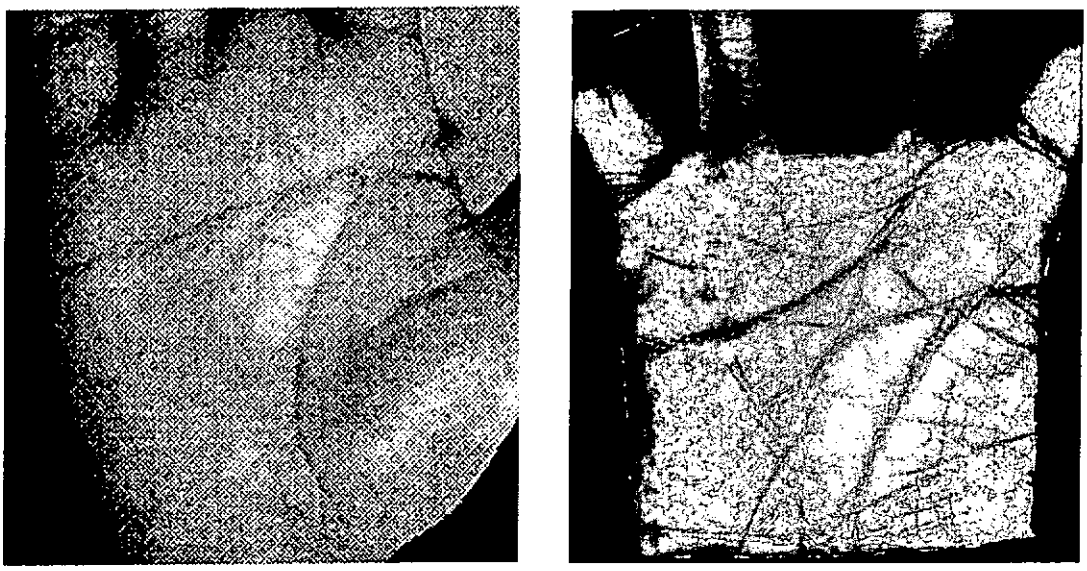
Fig. 4.4 Redesign the pegs on the flat platen surface for user comfort

4.2 The Appearance of the Device

The appearance of the device is a function of its size and shape yet while the shape of the device is determined by the type of sensor and its architecture (e.g. its optical path). There is a number of factors influence the size of a device. Firstly, user prefers a smaller device but the choice of capacitive sensor is constrained by its size which is incapable to acquire the entire area of a palm. We need to use a traditional optical system for the image acquisition. In the following section we describe the process of designing the optical system and the appearance of the device, from the first generation to the final design. The first two designs are made according to Niu's idea [10].

4.2.1 First Design – An L-Shaped Design

The first generation of our design is an L-shaped device with a mirror inside for reflecting light from the palm to the CCD sensor [10], (Fig. 2.4). The mirror would cut the optical axis horizontally and vertically thus reducing the height of the device but the mirror caused degradation of the image quality. Fig. 4.5 shows the palmprint image obtained by the first version acquisition device in (a), and the palmprint image obtained by the second version device in (b). The palmprint details in the image produced with the second version are much clearer.



(a) (b)

Fig. 4.5 The palmprint image obtained by the first version of the acquisition device in (a), and the palmprint image obtained by the second version device in (b).

4.2.2 Second Design – A Long-Tube Horizontal Design

A long-tube horizontal device was designed to address problems arising from the previous L-shaped design, as shown in Fig. 4.6. Since the most important issue in a palmprint acquisition device is the image quality, the optical system uses the traditional straight through optical axis. Here the light source used in the optical system is a light bulb. This approach greatly improves image quality.

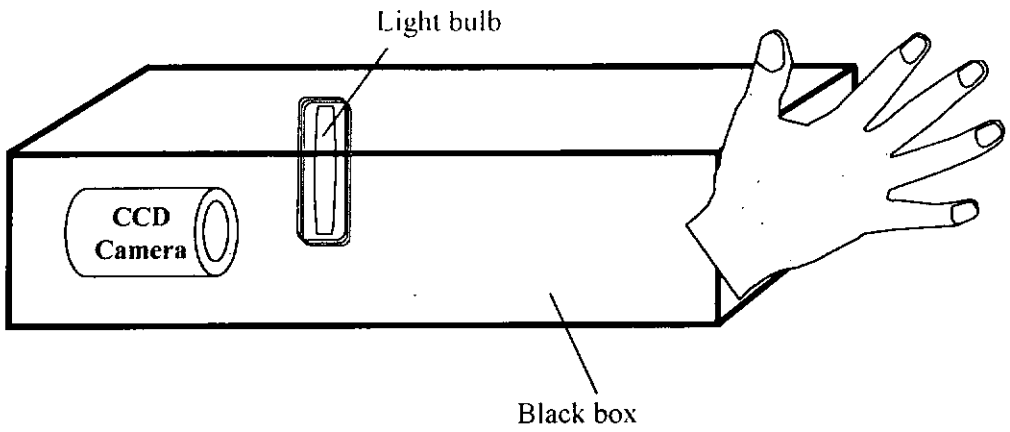


Fig. 4.6 Second version palmprint acquisition device: long tube horizontal device.

4.2.3 Third Design – A Long-Tube Vertical Design

Users found the horizontal placement of the device in the second design was inconvenient, so we made it upright while to ensure uniform illumination of the palm we changed the light source from a light bulb to a fluorescent ring light (Fig. 4.7 (a)). These changes support users in a more intuitive use of the device, and produce a more uniformly illuminated palmprint image.

4.2.4 Final Design – Enhanced Short Tube Design

Previous designs proved the lighting environment of the device to be effective so we were able to use a 6 mm focal length lens to reduce the height of the optical path from 240 mm to 130 mm. Further reduction in focal length could distort the captured image. The final version of the device was formed by applying the flat platen surface along with the best light source (ring fluorescent light), as shown in Fig. 4.7 (b). In our proposed palmprint acquisition device, we use a 1/2" lens format and a 1/3" CCD sensor format so that lens distortion near its rim is eliminated.

Having fixed the size and shape of the device, we applied the enhanced flat platen surface as the user interface, completing the design of the size, shape and user interface of the final palmprint acquisition device.

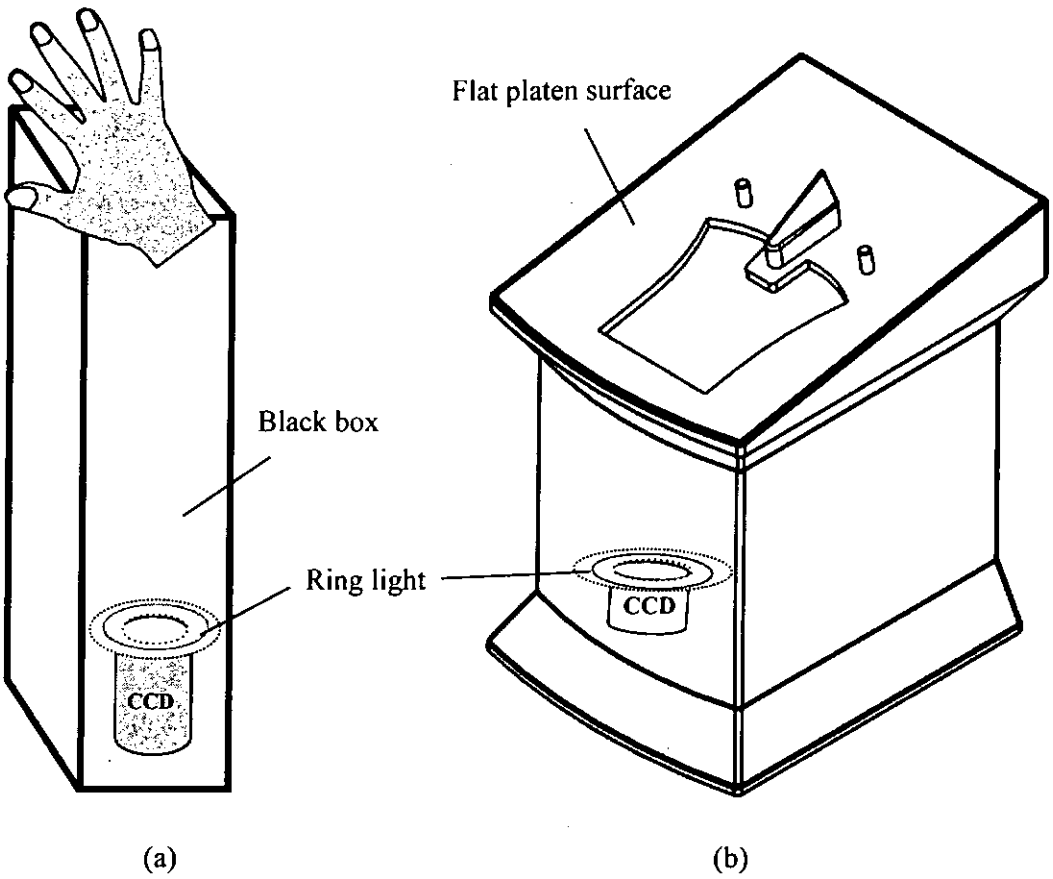


Fig. 4.7 (a) The third version of the palmprint acquisition device, (b) The final version of the palmprint acquisition device

5 EVALUATION OF THE ACQUISITION DEVICE

This Chapter evaluates the performance of our palmprint acquisition system and demonstrates how the requirements have been satisfied. Here we focus on:

1) resolution
2) uniform illumination
3) real time processing
4) image quality
5) effective palmprint pixels, and
6) real world example of our system

The evaluation result demonstrates how our system satisfied the requirements stated before. In fact, we used our system to acquire more than 9,400 palmprint images. We believe that the system is durable enough to be applied to real world applications such as building entrance control.

5.1 Resolution

Resolution refers to the *density* of pixels in a given linear area such as an inch. This "density" is expressed as ppi (pixels per inch) or dpi (dots per inch) [58, 62-66]. The term ppi (pixels per inch) originated in the world of computers, and dpi (dots per inch) in the world or printing, but today they are often used interchangeably.

5.1.1 Horizontal TV Lines (HTVL) Resolution

The picture sharpness and resolution of a CCD sensor is measured in HTVL [67]. Typical home-use CCDs have only 380 to 420 HTVL but their quality is far below from our requirements. Our system uses a black and white CCD sensor board which is a 1/3" Interline Transfer CCD, with 580 HTVL resolution, and electronic shutter speed is 10 μs.

There are 737 x 575 active pixels from an image sensor area of 4.8 mm x 3.6 mm where the pixel size is 6.6 μm x 6.3 μm. The maximum line pair per mm ($lpmm_{max}$) is calculated as [68]:

$$lpmm_{\max} = \frac{1}{2} p \,, \qquad (5\text{-}1)$$

where p is the pixel dimension.

A CCD sensor can transmit its maximum line pair only if a black bar falls on a pixel and a white bar on its neighboring pixel. Since the horizontal pixel size is 6.6 μm, the pixel dimension (p) is equal to 1000 μm / 6.6 μm = 151, so the maximum number of line pairs can transmit is (1/2) * 151 = 75 $lpmm$.

5.1.2 Spatial Resolution

One of the objectives of the proposed system is designed for real-time civilian applications. To fulfill the real-time requirements, high resolution images are not desirable since it creates a large image size and requiring more computation time for processing. We determined that our proposed palmprint acquisition device can be operated in medium resolution. The resolution (R) of the palmprint image is defined as:

$$R = Pw / Ow \,, \qquad (5\text{-}2)$$

where Pw is the width of the resultant image measured in points, and Ow is the width of the object (palm) in inches.

The target area is 5.1" x 3.8"; whereas the image formed is 768 x 576 points; the

image resolution is, therefore, 150 dpi. The image resolution can be further reduced to 75 dpi (1/2 of the original) by adjusting the parameters of the video frame grabber, i.e. the image size is 384 x 288 pixels for 75 dpi. As a result, we provide two different resolutions for researchers or application developers.

5.2 Evaluation of the Uniform Illumination

5.2.1 Design

We have been emphasizing that uniform illumination of the palm is very important in our framework. Thus, we performed an experiment to test different types of light source as shown in Figs. 5.1 (a) - (d), including:

1) Light bulbs
2) Light Emitting Diode (LED)s
3) Cold cathode fluorescent (CCF) lamps
4) Fluorescent lights

Since the LED is a point light source, we used forty white LEDs and arranged them into four groups, with ten LEDs per group on a bread board with electric wiring to provide enough illumination for capturing the palmprint image. For the CCF lamp and fluorescent light, we used a ring shape design to provide better uniform illumination of the palm.

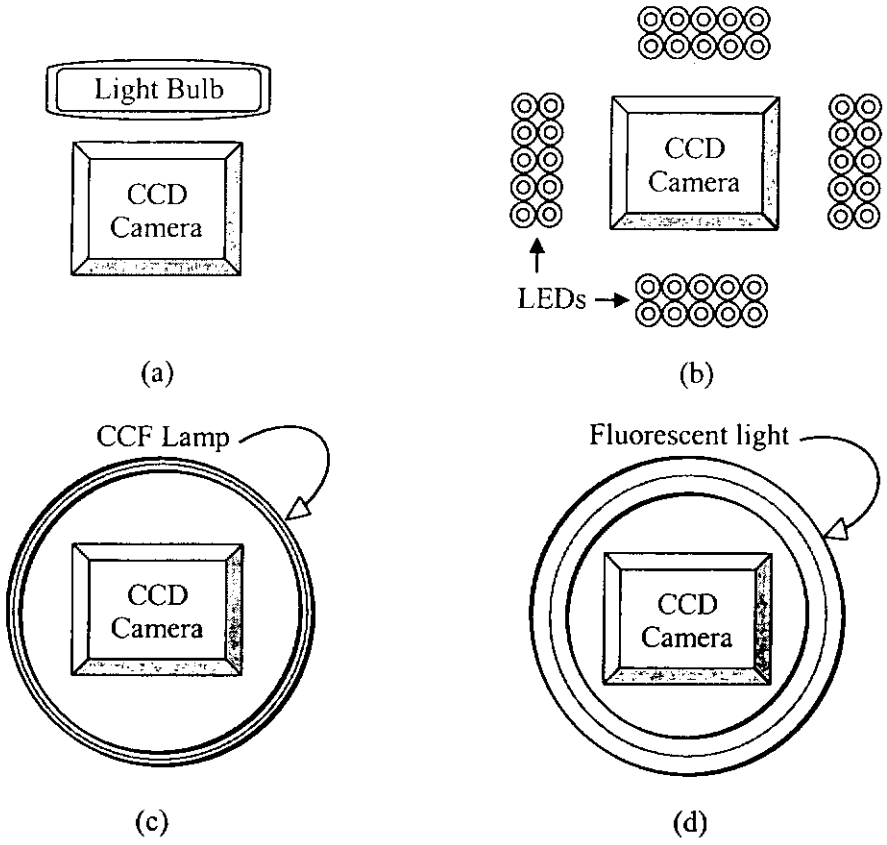


Fig. 5.1 Different light sources: (a) light bulb, (b) LEDs, (c) CCF lamp, and (d) fluorescent light.

5.2.2 Experiments

We set up a database of 20 hands from 10 volunteers for the testing. Fig. 5.2 shows the experimental results on uniform illumination of the palm by: (a) light bulb, (b) LEDs, (c) CCF lamp, and (d) fluorescent light. Table 5.1 compares the palmprint images in terms of illumination, contrast and sharpness under different light sources. We found that the fluorescent light provides the best result. CCF lamp produces the second best result, while the light bulb recorded the worst: there is uneven illumination on the palm, causing a brighter lower left part and a dimmer upper right part on the palm. For the LEDs situation, the palm was badly illuminated. We chose the ring shape fluorescent light as our light source with reference to the above observations, ensuring uniform illumination of the palm.

Table 5.1 Image quality obtained using different light sources.

	Uniform illumination	Contrast	Sharpness
Light bulb	Bad	Acceptable	Good
LEDs	Fair	Acceptable	Fair
CCF lamp	Good	Acceptable	Acceptable
Fluorescent light	Good	Good	Good

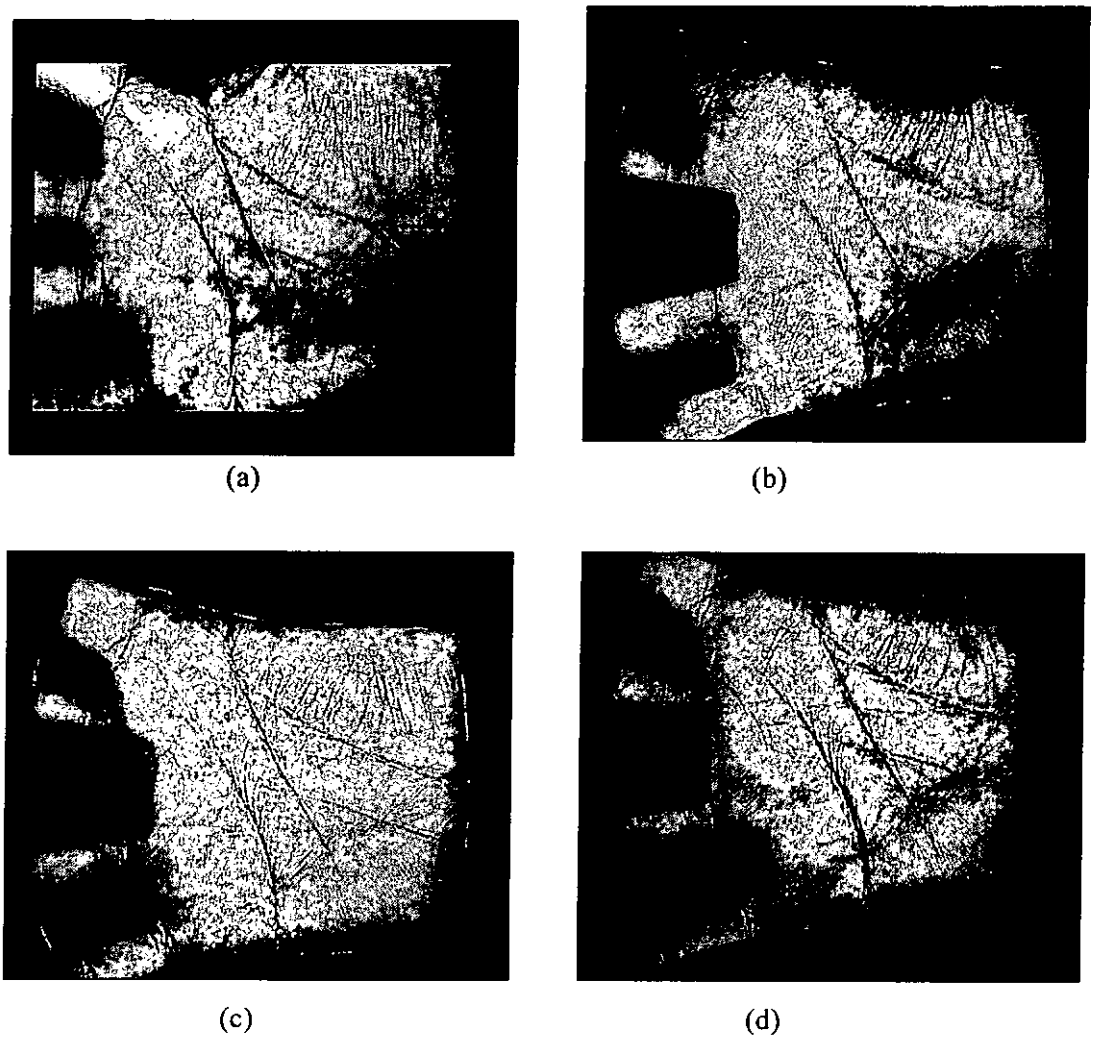


Fig. 5.2 Experimental results on uniform illumination of the palm by: (a) light bulb, (b) LEDs, (c) CCF lamp, and (d) fluorescent light.

5.3 Evaluation of Real Time Processing

We wrote a program that would grab a frame of a palmprint image from our proposed device and to transfer it to the main memory. The time for capturing an image is calculated by: *time to obtain one frame* + *time to write it to the main memory* + *latency time of the CPU*. The time to obtain one frame is 1 *second* / 25 *fps* = 0.04 *second*. We set up an experiment to calculate the time taken from the start of the acquisition process until the image is transferred to the main memory.

We performed the experiments using three different speed CPUs that were connected to a standard hard disk. The total time to store a palmprint image is 0.22s, 0.11s and 0.09s on CPUs of Pentium 233 MHz, Celeron 950 MHz and Pentium 4 1.8 GHz respectively. Since the Pentium 233 MHz CPU is installed on an industrial board with an extra disk-on-chip module, we performed an extra test using DiskOnChip 2000®, and the time used was 0.11s, half of that of using a normal hard disk. This implies that the disk speed is also an important factor in image acquisition speed. We can say that our system can obtain a palmprint image in real time, even operating in a slow computing environment. On the other hand, for comparison purpose, we acquired a grayscale image from a scanner with resolution of 150 dpi using a computer running on Windows XP with Pentium 500 MHz CPU. It took ten seconds for acquiring an image of 874 x 880 in size.

5.4 Evaluation of the Image Quality

The evaluation of image quality is critical to an acquisition system. We tried to analyze how the palmprint features was acquired by our system. The image quality is independent of algorithm, i.e. the system is not designed for a special algorithm for palmprint matching. In view of this circumstance, we do not intend to associate the image quality with the accuracy of any algorithm used for palmprint identification. There are rich

features found in a palm. They are classified into three levels: principal lines, wrinkles and ridges [31]. As a matter of fact, different researchers would like to use different palm features for their researches. However, we would like to show that our device is able to obtain every feature available in a palm.

Fig. 5.3 shows a palmprint image obtained using our device at 150 dpi with enlargement in two highlighted areas, depicting the three-level features. Fig. 5.4 (a) shows a palmprint with different resolutions: 75 dpi, 100 dpi, 125 dpi, and 150 dpi. All the three levels of line features can be obtained using our system at 150 dpi. The finest level of features (ridges) started to blur as the resolution decreased. Based on the collected 9,400 images, we summarized the findings on the resolution requirements for different levels of features from a palm in Fig. 5.4 (b). According to the experiment results, it suggest that all those three levels of line features can be obtained using resolution from 125 dpi to 150 dpi, whereas the resolution from 100 dpi to 125 dpi can only get wrinkles and principal lines. For a resolution of less than 100 dpi, only principal lines are still clear that is favorable for the feature extraction. More different palmprint images with size of 420 x 420 cropped from the original images are illustrated in Fig. 5.5.

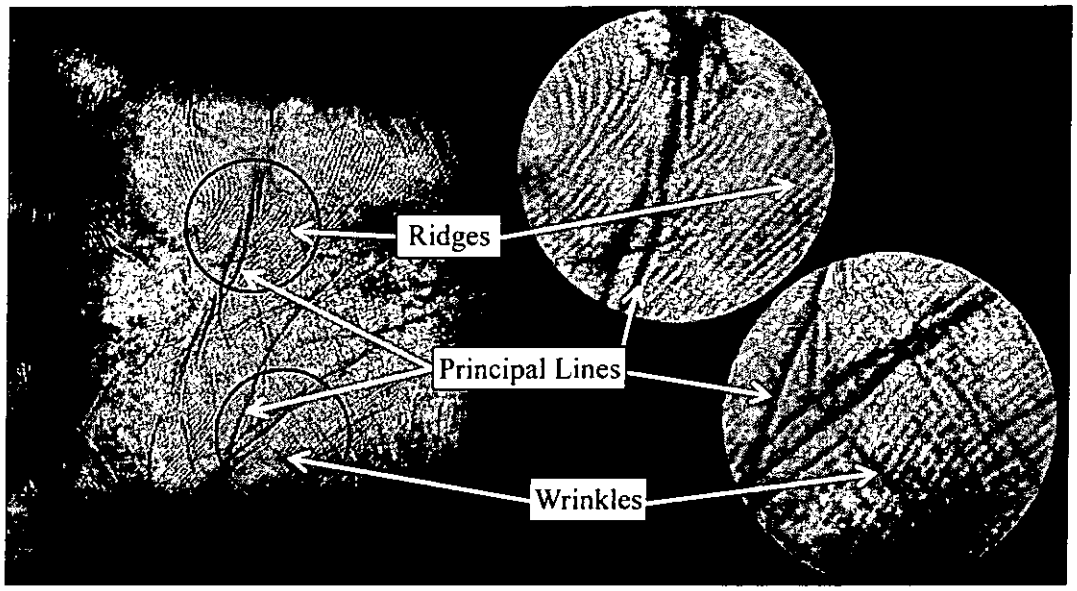
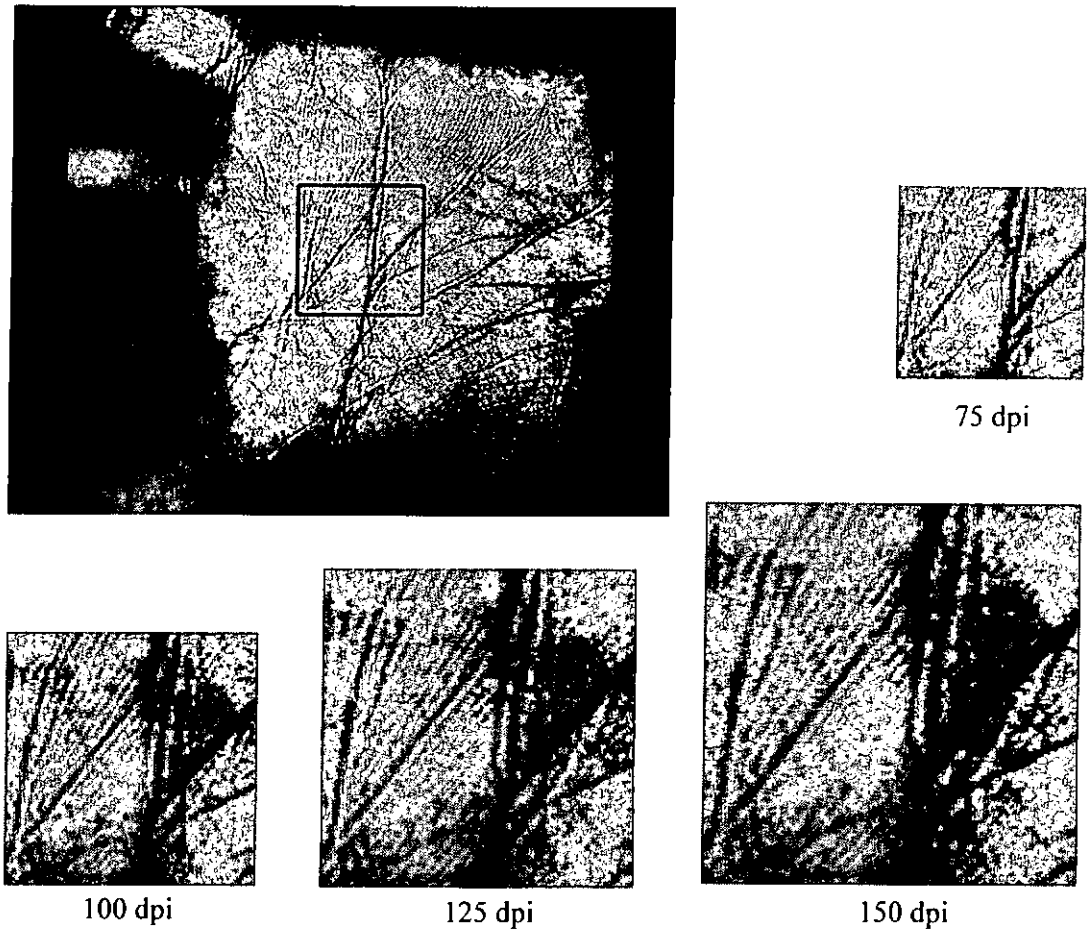
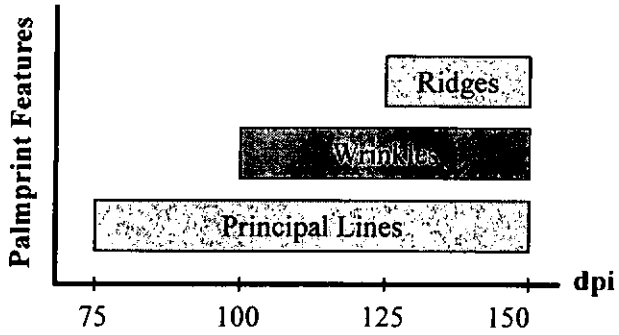


Fig. 5.3 Palmprint images obtained using our device at 150 dpi.



(a)



(b)

Fig. 5.4 A palm with different resolutions at (a): 75 dpi, 100 dpi, 125 dpi, and 150 dpi. (b) The resolution requirement for different levels of features from a palm.

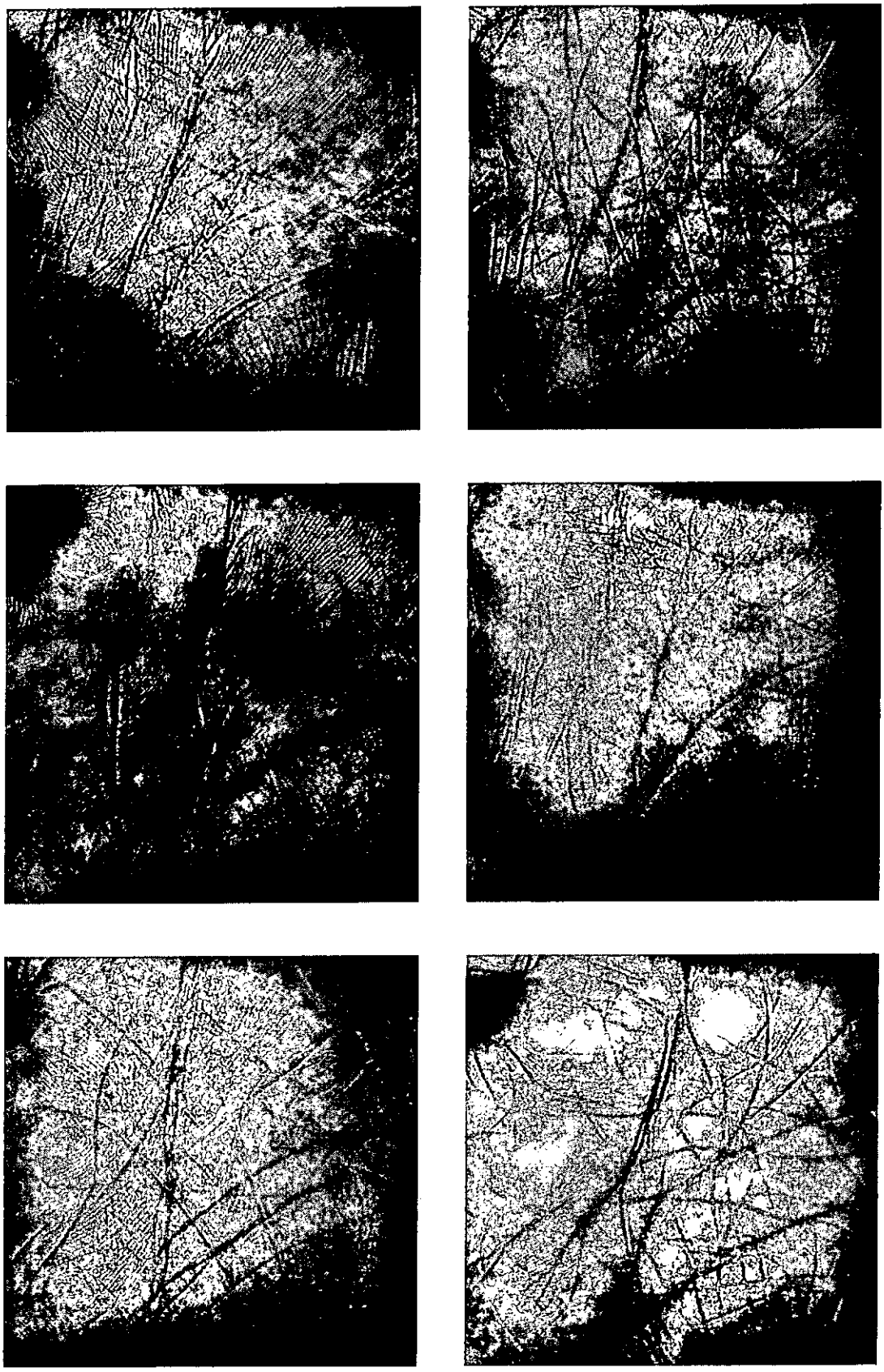


Fig. 5.5 Different palmprint images with size of 420 x 420 cropped from the original images.

5.5 Evaluation of the Effective Palmprint Pixels

In an ideal situation, our system obtains the palmprint data (functional for palmprint feature extraction) equivalent to 420 x 420 pixels from the flat platen surface. However, some non-palmprint pixels are also acquired from the system due to a hand with big rotation or a hand with smaller size, leading to a smaller extracted palmprint.

We carried out an experiment to examine effective palmprint pixels which can be extracted from each palmprint image. The purpose of the experiment is to found out the suitable size of a palmprint for the feature extraction. The database has a total of 9,400 images obtained from 235 individuals using our palmprint acquisition system. The subjects are mainly students and staff volunteers from The Hong Kong Polytechnic University, 157 male and 78 female, 10% younger than 20 years old, 78% 20-29 years old and 12 % older than 30. The experiment results are summarized in Figs. 5.6.

If a palm completely covers the central part of the flat platen surface (the area enclosed by the white box shown in the figure), our system is able to get a full-size of 420 x 420 pixels of palmprint data, as shown in Fig. 5.6 (a). However, we could only get as much as 318 x 318 pixels (a number smaller than the expected values) from a child's hand, as shown in Fig. 5.6 (b). From Fig. 5.7, it can be seen that by using a 420 x 420 pixels area, only 75% of palms can obtain palmprint data without non-palmprint pixels. On the other hand, by using a 400 x 400 pixels area, more than 95% of palms can obtain palmprint data without non-palmprint pixels. The difference is shown in Fig. 5.6 (c) where the outer white box is the extraction area equivalent to 420 x 420 pixels, containing some non-palmprint pixels that occurred on the top and the left of the image. In the inner yellow box, an area of 400 x 400 pixels, we obtained no non-palmprint pixel. This experimental result suggests that our device is effective on obtaining palmprint data at 400 x 400 pixels with more than 95% of palms without non-palmprint pixels.

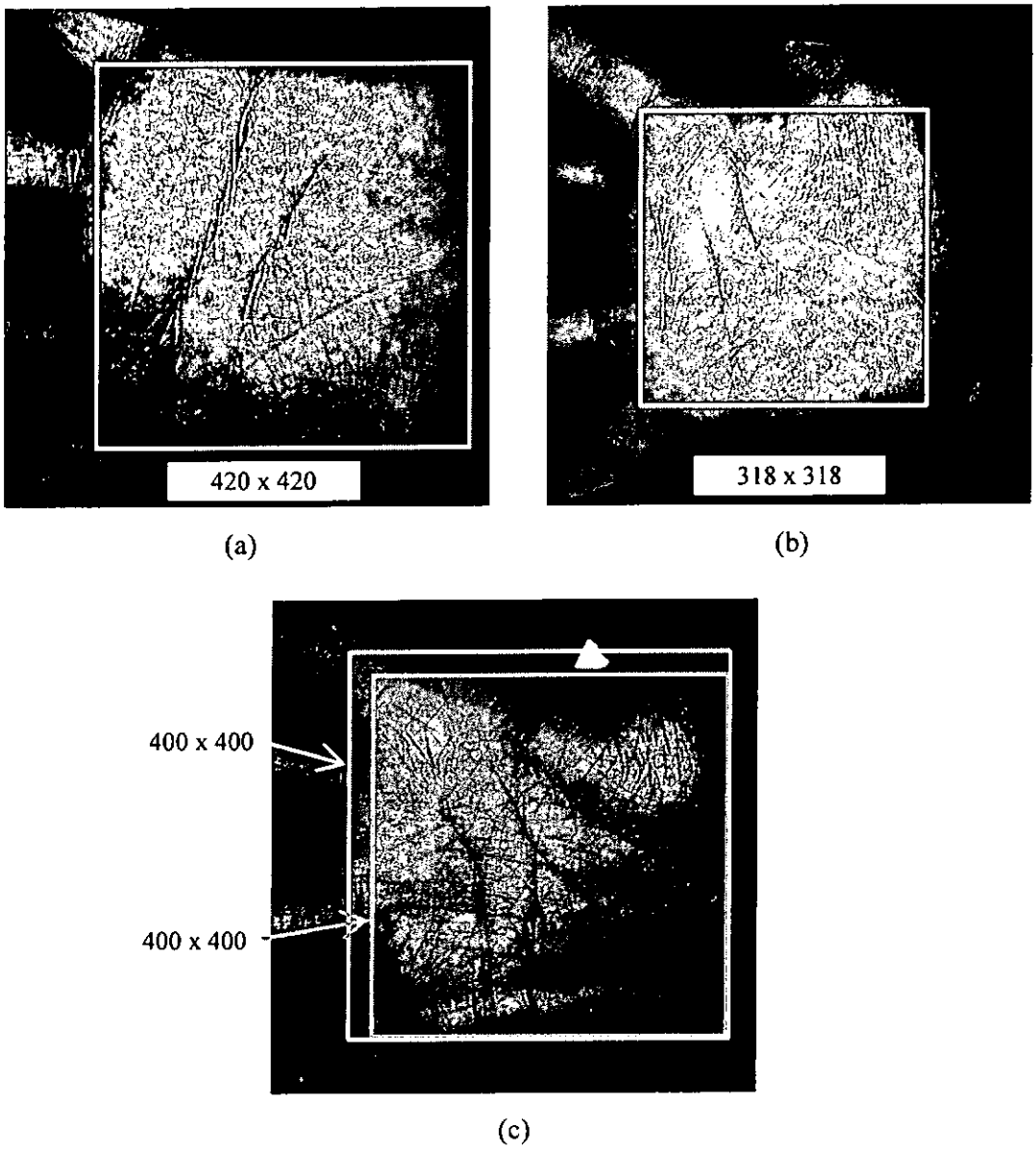


Fig. 5.6 Palmprint data extraction: (a) an adult's palm, (b) a child's palm, (c) difference between extraction area of 400 x 400 and 420 x 420.

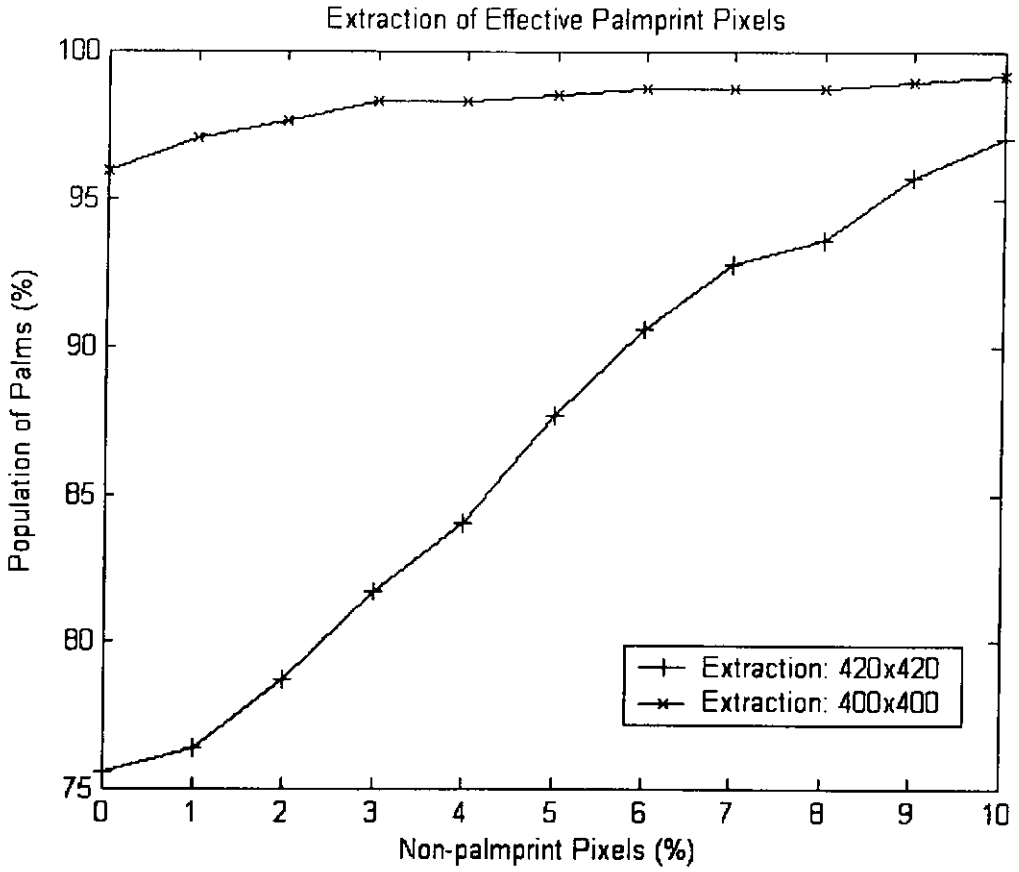


Fig. 5.7 Experimental results on the extraction of effective palmprint pixels from different palms.

5.6 Real World Example

To test the feasibility of our palmprint acquisition system, we connected it to a recognition engine developed by us [43]. We put the palmprint acquisition system with the recognition engine together to form a palmprint identification system as seen in Fig. 5.8, which is placed at our laboratory entrance for access control since early March 2003. All the personnel inside the laboratory were enrolled in the system at the beginning. We have been using our palm as the 'key' to enter the laboratory. So far, the system has been running non-stop for more than ten months without any problems.

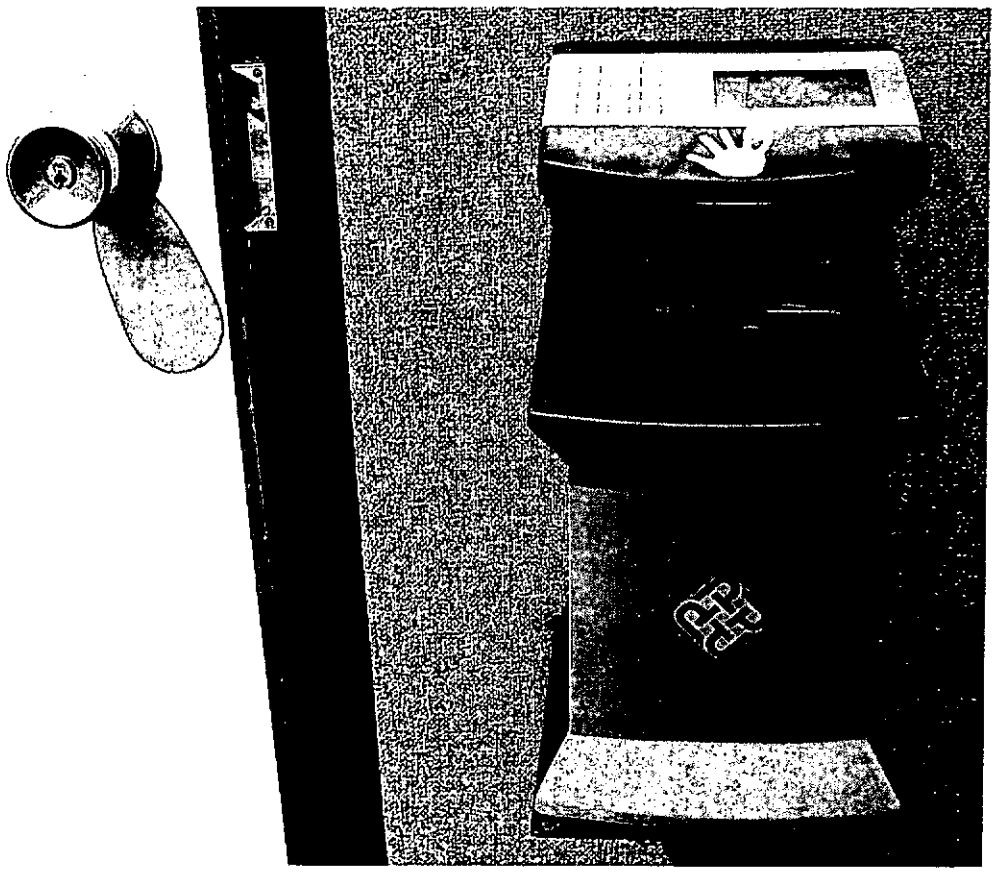


Fig. 5.8 A palmprint identification system at entrance our own laboratory for access control.

Part III: Effective Palm Line Extraction

6 EXISTING METHODOLOGY AND PROPOSED METHOD

A biometric identification/verification system includes stages of signal acquisition, preprocessing, feature extraction and matching. Palmprint identification system is no exception. In the signal acquisition stage, a palmprint signal is acquired by an image sensor during enrollment or identification procedure. In-depth details of the design and implementation of a palmprint acquisition device have already been discussed in Part II. This Part is going to discuss the recognition engine based on the line feature extraction and matching. It involves palmprint signal preprocessing, line feature extraction, and line matching. There are four chapters in this Part, which guides through the whole architecture of the proposed method of the recognition engine. In this Chapter, we first give a literature review on the existing methodology, including palmprint signal preprocessing, feature extraction, and matching, proposed by different researchers. Next, we give some introduction to our proposed method. In Chapter 7, we discuss the proposed idea on the palmprint signal preprocessing, which involves binarization and edge detection on the palmprint image. In Chapter 8, a line feature extraction algorithm is discussed by applying knowledge from the palm line structures. We defined rules and strategies to enhance the performance of correct line feature extraction. Then, the proposed matching strategies are discussed in detail in Chapter 9. The performance of our

proposed method including the experimental results is also revealed there.

6.1 Palmprint Signal Preprocessing

The main purpose in this stage is to obtain a sub-palmprint image with rotation and translation invariance for feature extraction. It is important to define a coordinate system to align different palmprint images. There are different approaches to obtaining the coordination system [31, 45, 48, 50, 69], described in the following Sections.

6.1.1 Shu et al. Method

This coordination system is determined by using end points *A* and *B* from the heart line and life line of a palm [31, 48], see Fig. 6.1. If we can accurately determine points *A* and *B*, these points can provide a relative stable coordination system for palmprint feature extraction. However, according to Shu's experimental results, if life line and head line are not intersected, the accuracy rate only achieve 80% whereas rotated and incomplete image achieved even better results at 94% and 95% respectively. This suggests that the accuracy of this coordination system can be affected by the intersection of those lines.

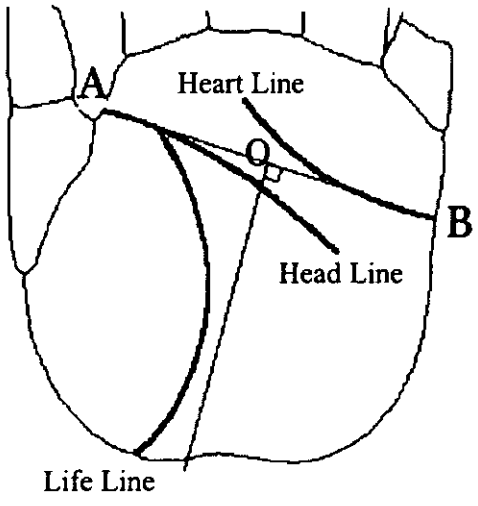


Fig. 6.1 A coordination system proposed by Shu et al.

6.1.2 Li et al. Method

This coordination system is determined by using the holes created by fingers during the image acquisition process [69]. There are three finger holes created on the image which provide tracking paths on the coordination of the palm, see Fig. 6.2. By the calculation on the centre of gravity from boundaries of holes, we can obtain three key points, *k1*, *k2*, and *k3*. Then, make a line from *k1* to *k3*, to form the Y-axis of the palmprint coordination system. Similarly, make another line perpendicular to the Y-axis through *k2* to determine the origin of the palmprint coordination system. Having established the coordination system, we can locate a fixed part of sub-image, called "central part sub-image", for feature extraction.

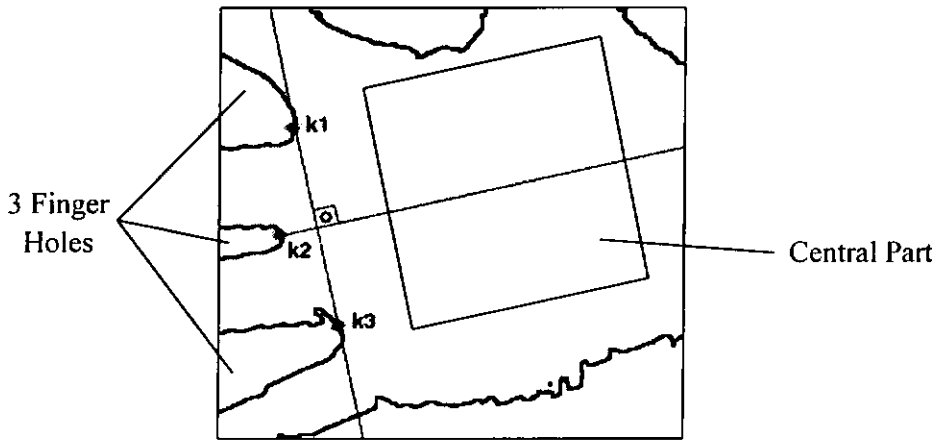


Fig. 6.2 A coordination system proposed by Li et al.

6.1.3 Han et al. Method

This coordination system is mainly determined by the boundary traced from the middle finger. The algorithm first locate the end points *a*, *b* and *c* as shown in the Fig. 6.3. Then, it joins point *a* and *b* to form a base line and draw a perpendicular line from point *c* through the mid-point of *a* and *b* as the origin [45]. This coordination system depends heavily on the positioning of the middle finger. A slight change in the orientation of the middle finger may greatly affect the result of the sub-image extraction.

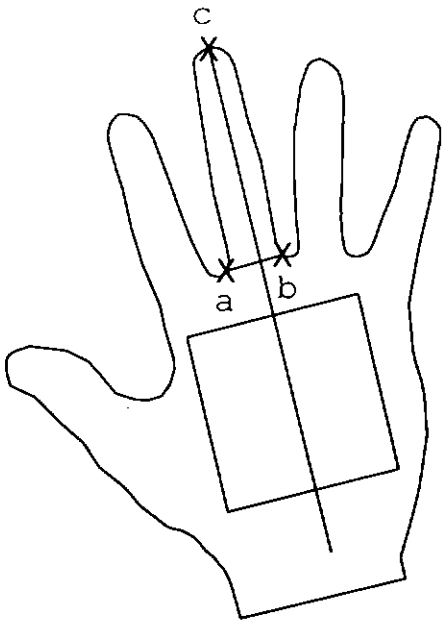


Fig. 6.3 A coordination system proposed by Han et al.

6.1.4 Kong's Method

There is a specialized device for the acquisition of palmprint image where there are three pegs on the interface to guide the placement of hand. The specialized device provides a comparatively stable environment for the acquisition of a palm image. There are two finger holes formed automatically by a palm covering the cropped area from the flat platen surface, as shown in Fig. 6.4 (a). Then, a list of points can be obtained by searching for the finger holes from the binary image. Actually, these two finger holes are key to the determination of the palmprint coordinate system [50]. The major steps of this coordination system are as follows:

Step 1: Binary the image

Use a thresholding technique to convert the original grayscale image to a binary image, and then use a low-pass filter to smooth the binary image, see Fig. 6.4 (a).

Step 2: Trace the Boundaries of finger holes

Trace the boundaries of holes *H1* and *H2* to obtain two lists of points. *H1* is a hole created between the ring and little fingers, while *H2* is a hole created between the index

and middle fingers, as shown in Fig. 6.4 (b).

Step 3: Compute the tangent points T1 and T2.

Tangent points *T1* and *T2* were calculated from the holes *H1* and *H2* respectively, as shown in Fig. 6.4 (c).

Step 4: Establish the coordination system

Draw a line pass through *T1* and *T2* to form the Y-axis of the palmprint coordination system. Then make another line pass through the midpoint of *T1* and *T2*, which is perpendicular to the Y-axis to form the origin of the coordination system, see Fig. 6.4 (c).

Step 5: Central part extraction

Extract a fixed size of sub-image from the coordination system, see Fig. 6.4 (d).

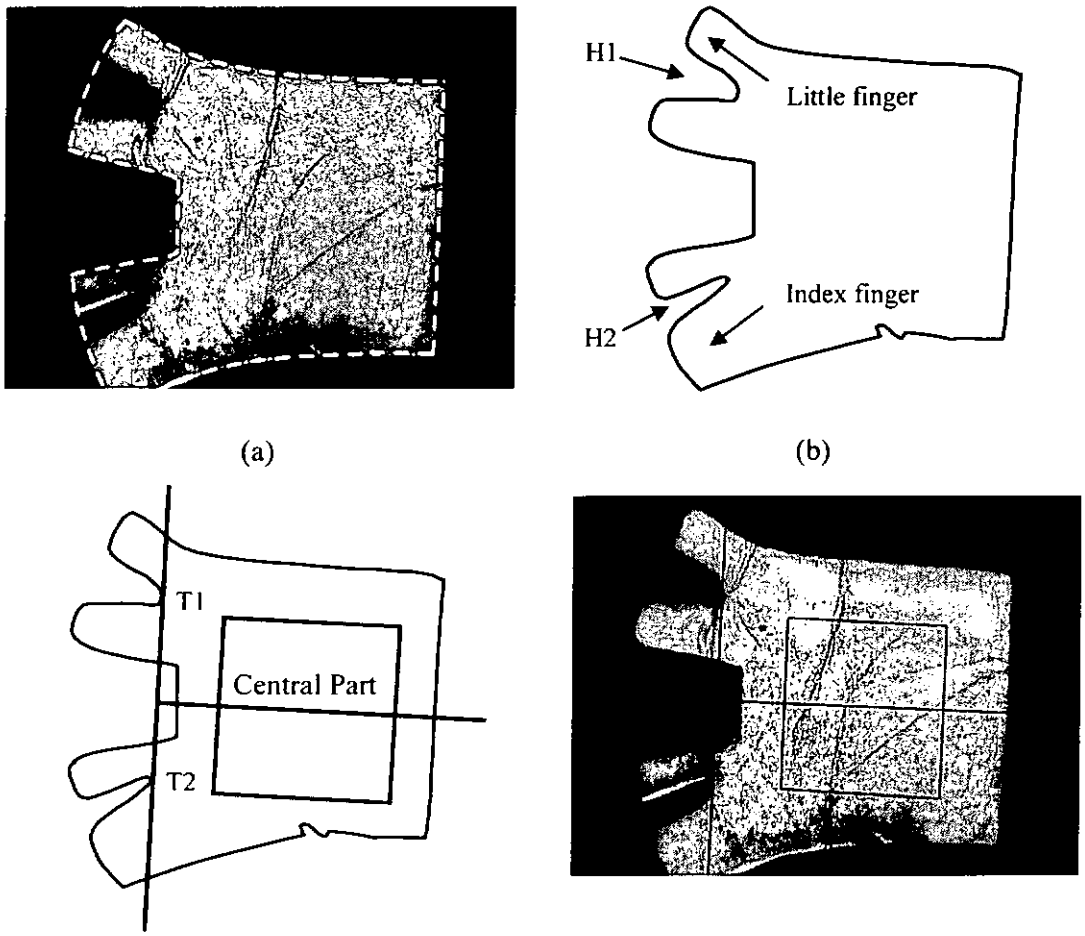


Fig. 6.4 Kong's coordination system. (a) Original image, (b) boundary tracking, (c) coordination system established, and (d) central part of a palm.

6.2 Feature Extraction and Matching

In the feature extraction stage, unique features are extracted from the palm to create master templates for matching. There are three levels of features found in a palm: principal lines, wrinkles and ridges. Our device established in previous Chapters obtains these features in 150 dpi accordingly, as shown in Figs 5.3 – Figs. 5.5. Some researchers use the whole palm to get the features while some researchers use only the central part sub-image of a palm. The selected features should possess the properties of low intra-palm difference and high inter-palm difference. The coming Sections describe different approaches on palmprint feature extraction and matching schemes, proposed by different researchers [31, 48, 50, 70].

In the matching stage, the identification template is compared with the master templates such that the system determines whether they are from the same person. Different approaches to feature extraction directly affect the types of methods used in the matching process.

6.2.1 Shu's method

Feature Extraction

Palmprint data is obtained offline through the scanning on the paper of inked palmprint. Shu [31, 48] uses short straight line segments as the palmprint feature because they think that curvature of the principal lines is small enough to be represented by several short straight line segments. They use four different directional templates (vertical, horizontal, left and right diagonal) for line segment determination. Then they post process the results on each directional template. After that, they combined all four directions results and repeat the post processing to eliminate some overlapped line segments.

Matching

Since the features are made up of short line segments, the matching process is performed

by calculating the Euclidean distances and angle of inclination between two lines. If both distances and angles are less than a given threshold, the two lines are regarded as the same. Their experiment results show that the accuracy rate on rotated image is 95%, for incomplete images is 94% whereas the life line and head line unintersection is only 80%.

6.2.2 You's method

Feature Extraction

You [70] proposed an algorithm that combines the global and local palmprint feature providing dynamic selection scheme to facilitate the coarse-to-fine pattern matching. Global Texture Energy (GTE) is used to represent the global palmprint feature which is characterized by high convergence of inner-palm similarities and good dispersion of inter-palm discrimination. They borrowed the idea of Law's four most powerful masks which is believed to be sensitive to visual structure such as edges, ripples and spots [71]. Then, they use adaptive masks which are better for texture features and more sensitive to horizontal line, vertical line, 45 degree angle line and 135 degree angle line.

At a fine level, they used an interest point based method rather than an edge based method for the feature representation. They defined the interesting points which are distinctive and representative from each palmprint image by using a Plessey operator [72].

Matching

The matching process is defined by using a Hausdorff distance algorithm [73] to search for portions or partial hidden object. It is a non-linear operator which determines the degree of mismatch of the two sets by measuring the distance of the point of a model that is farthest from any point of an object and vice versa. It can be used for palmprint recognition by comparing two images which are superimposed on one another. The average accuracy rate is 95%.

6.2.3 Kong's method

Feature Extraction

Kong [50] proposed a texture based algorithm for the feature extraction, which treats the palmprint as a piece of texture. If the algorithm only uses the principal lines as the features, it can only provide very low discrimination power as many people have similar principal lines. Wrinkles provide plentiful of features which increase the discrimination power and can be used to uniquely identify a person. However, it is difficult to accurately extract wrinkles. Kong introduced a texture based method using Gabor filter for the palmprint feature extraction. The circular Gabor filter is an effective tool for texture analysis [17] with the following general form,

$$G(x,y,\theta,u,\sigma)=\frac{1}{2\pi\sigma^2}\exp\left\{-\frac{x^2+y^2}{2\sigma^2}\right\}\exp\{2\pi i(ux\cos\theta+uy\sin\theta)\}, \qquad (6\text{-}1)$$

where $i=\sqrt{-1}$; u is the frequency of the sinusoidal wave; θ controls the orientation of the function and σ is the standard deviation of the Gaussian envelope. In order to provide more robust to brightness variability, a discrete Gabor filter, $G[x, y, \theta, u, \sigma]$, is turned to zero DC with the application of the following formula:

$$\tilde{G}[x,y,\theta,u,\sigma]=G[x,y,\theta,u,\sigma]-\frac{\sum_{i=-n}^{n}\sum_{j=-n}^{n}G[i,j,\theta,u,\sigma]}{(2n+1)^2}, \qquad (6\text{-}2)$$

where $(2n+1)^2$ is the size of the filter. In fact, the imaginary part of the Gabor filter automatically has zero DC because of odd symmetry.

Matching

Each feature vector would be considered as two 2-D feature matrices, real and imaginary part. The matching score will be determined by a normalized hamming distance. For a 1-to-100 identification, their system can operate at 0.1% false acceptance rate with 97% genuine rate. For a one-to-one verification mode of operation, their system can operate at 98% acceptance rate with 0.04% false acceptance rate.

6.3 The Architecture of Our Proposed Method

There are four stages in our proposed method, as shown in Fig. 6.5. The first stage is the palmprint acquisition, which was discussed thoroughly in Part II of the thesis. The remaining three stages will be discussed in the coming chapters, they are: 1) palmprint signal preprocessing, 2) line feature extraction, and 3) line matching.

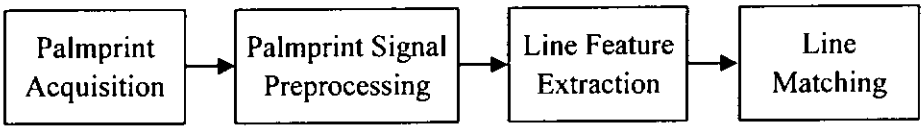


Fig. 6.5 Four stages of our proposed method.

1. Palmprint Signal Preprocessing

When a raw palmprint image is acquired by the sensor, usually, there are some noises added to the image. We need to perform some image processing techniques to enhance it. We proposed to use the adaptive thresholding method to get a preliminary line map of a palmprint image in spite of there are a lot of edge detectors. Having obtained the line information of the image, we applied thinning operation on the image to get the one pixel width palm line features. Details are shown in Chapter 7.

2. Line Feature Extraction

When the palmprint image has binarized and thinned, it is ready to perform the palm line extraction process. However, noises are usually added to the thinned palmprint image which cause problem in the correct line extraction. We here proposed to use a knowledge based method on the palm line extraction from the preprocessed image. By incorporating rules from the knowledge of the palm line structures, the performance of correct palm line extraction is enhanced. Details are shown in Chapter 8.

3. Line Matching

In this stage, the line features from the identification template is compared with the

master templates such that the system determines whether they are from the same person. The line features are stored as a set of points in sequence. We first group every six points into a node, and then we compare the angle for each node to determine a score. If the angle is less than the threshold, we add the score and the number of comparisons; otherwise, we only add the number of comparisons. Finally, we get the matching score by dividing the total score by the number of comparisons. Details are shown in Chapter 9.

7 PALMPRINT PREPROCESSING

In the palmprint signal preprocessing stage, some image enhancement procedures are performed to reduce noises. Here we applied a lowpass filter (i.e. Gaussian) to the original image to smooth it. Next, a coordination system is established to locate the palmprint data from the raw image. We adopted the coordination method proposed by Kong for the central part extraction since it has concerning the stable positioning of palm by the design of user interface. Then, we need to do the binarization of the image to get the preliminary line map of a palmprint image. Although different filters are designed for edge and line detection, such as Laplacian of Gaussian (LoG) and Canny edge detector, they are not effective on the palm lines extraction since they are all dependent on the global thresholding technique which is not good for the peculiarity line features of a palm. Detail discussions on traditional edge detection techniques are presented in Section 7.1. We proposed to use the adaptive thresholding method to get the preliminary line map of a palmprint image as it gets better result. Next, we perform thinning operation on the line map to get the one pixel width palm line features. Detail discussions on the thresholding techniques are reported in Section 7.2 and the conclusions are given on Section 7.3.

7.1 Edge Detection Techniques

Edge detection is the fundamental process for different image processing, image understanding and pattern recognition applications [74]. Edges characterize boundaries and refer to the regions which have abrupt changes of grayscale values hence they can be extracted by detecting the changes. Gradient magnitude describes the strength of an edge, while the gradient orientation indicates the direction of the edge. However, there is no universal algorithm that is best for all kinds of images. Generally, an algorithm has a set of parameters, which needs to be optimized by experiments to fit for a specific application. There are different filters designed for the edge and line detection, including: Zero Crossing Detector, Laplacian / Laplacian of Gaussian (LoG), Sobel edge detector and Canny edge detector, etc. The following sub-sections describe the principals of Sobel, LoG, and Canny, and performed experiments using these edge detectors.

Figs. 7.1 show the palmprint images which have: (a) full of wrinkles, (b) very strong principal lines and strong wrinkles, (c) very weak and blurred lines, and (d) clear principal lines. These images are the central part sub-image obtained by using Kong's algorithm. All the experiments performed in the coming Sections are using these same sets of images.

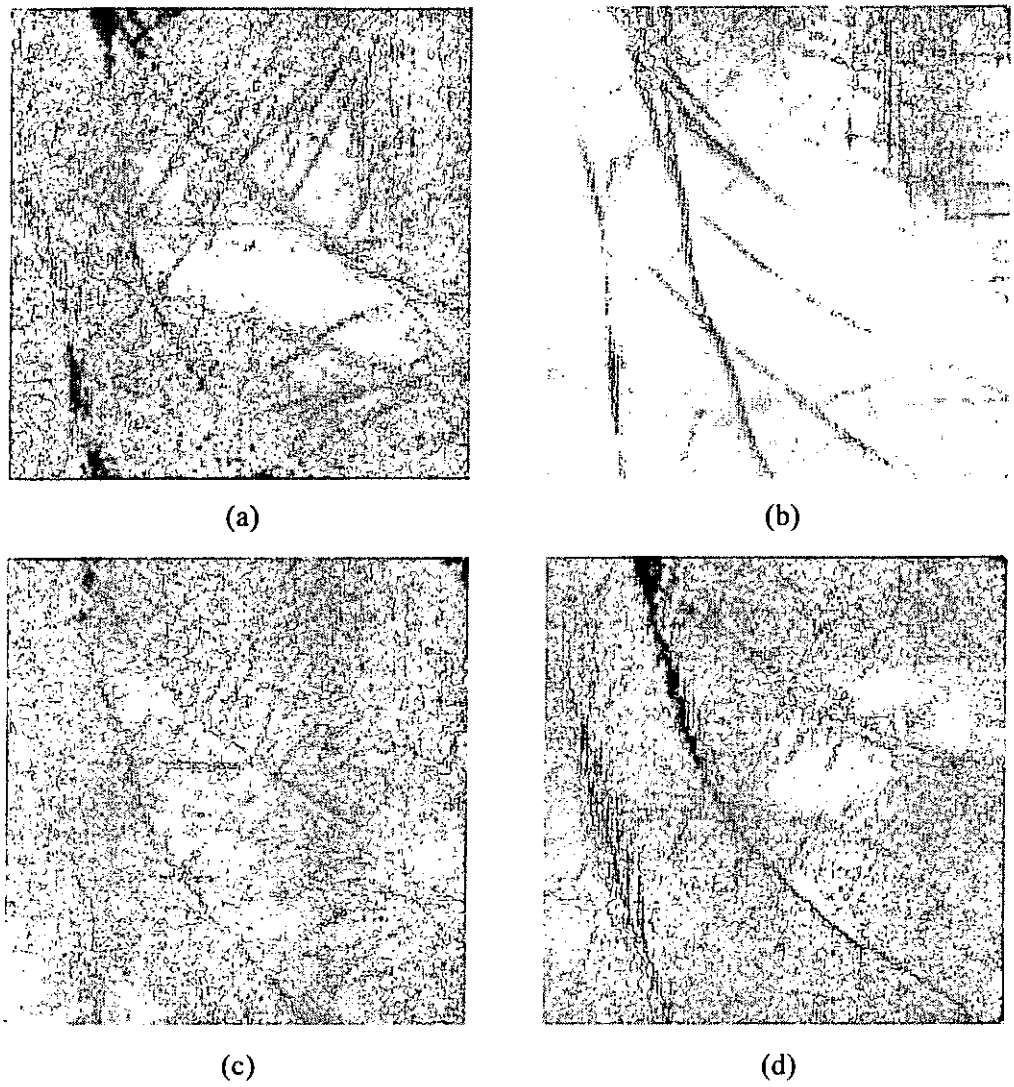


Fig. 7.1 Central part sub-image of different palmprints.

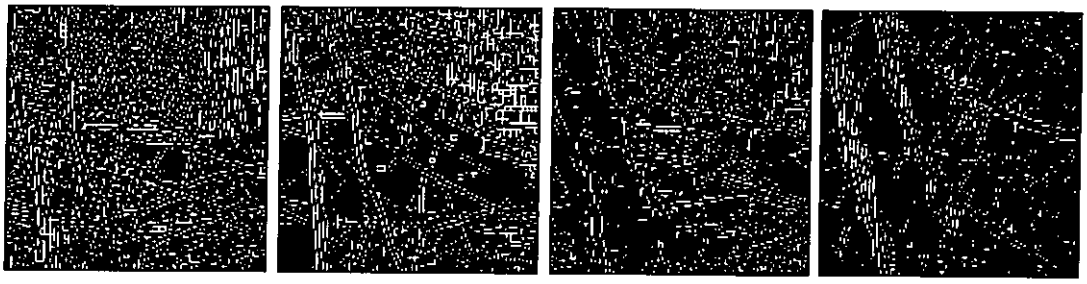
7.1.1 Sobel

The Sobel edge detector makes use of a pair of convolution kernels. One is used to detect vertical edges and the other is for horizontal edges. The convolution results from the horizontal and vertical Sobel edge detectors are then combined to form the absolute gradient magnitude of the image [75].

Experimental Results

The source images are shown in Figs. 7.1, while results are shown in Figs. 7.2, generated by MATLAB 6.1. We have tuned the best threshold such that the output is the most

appropriate for comparison. The threshold values used in Figs. 7.2 are: (a) 0.022, (b) 0.015, (c) 0.025, and (d) 0.03. It seems that those lines extracted are messy with no clear line structures formed.



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

Fig. 7.2 Results obtained by Sobel edge detector.

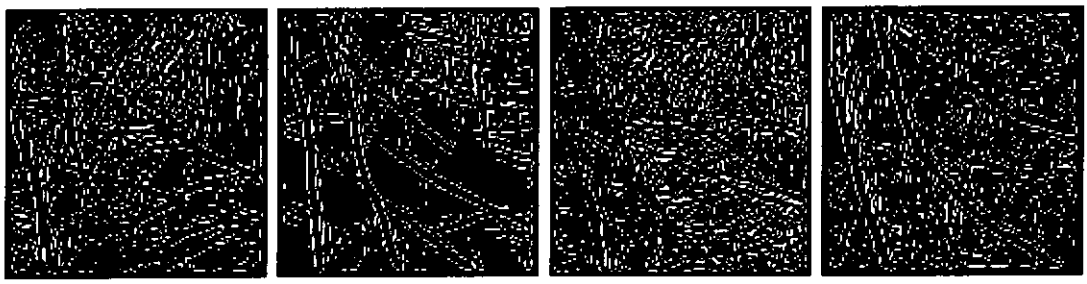
7.1.2 LoG

Laplacian of Gaussian (LoG), on the other hand, is the second derivative of Gaussian that tries to balance the noise suppression and edge detection. It is common to smooth the image by a Gaussian filter before applying the Laplacian. LoG is the combinations of the two operations into one single operator. [76]

Experimental Results

The source images are shown in Figs. 7.1, while results are shown in Figs. 7.3, generated by MATLAB 6.1. We have tuned the parameters of the threshold and sigma such that the output is the most appropriate for comparison. The threshold (t) and sigma (δ) values used in Figs. 7.3 are: (a) t = 0.008 and δ = 1.4; (b) t = 0.008 and δ = 1.4; (c) t = 0.008 and δ = 1.2; (d) t = 0.004 and δ = 1.5.

There are some double lines which are enclosing the palm lines because of its zero crossing characteristics. For those weak palm lines images, the output is come to a chaos and difficult to differentiate the lines from the zero crossing points.



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

Fig. 7.3 Results obtained by LoG edge detector.

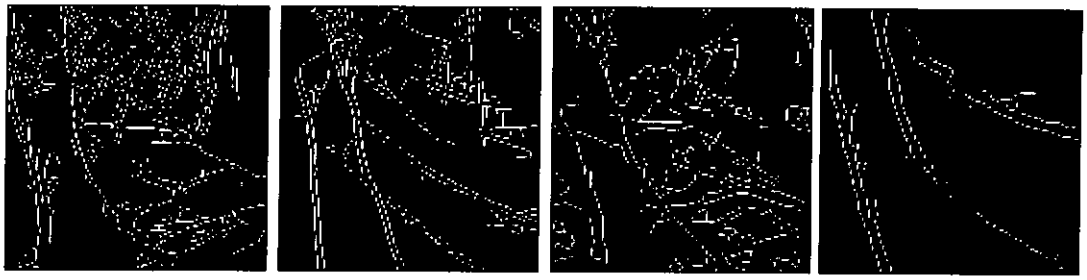
7.1.3 Canny

Canny edge detector [77] is one of the most famous edge detectors in the field of image processing. It takes a grayscale image as input and produces an output with the tracks of intensity discontinuities. The image is first smoothed by Gaussian filter, and then a simple 2D first derivative operator is applied to the smoothed image to highlight regions with high spatial derivatives from the image. After that, it tracks along those regions and suppresses any pixels that are not at the maximum called non-maximal suppression. This process was controlled by two thresholds, *T1* and *T2*, where $T2 > T1$. Tracking only begin at a point on edge higher than *T2*, and then continues in both directions from that point until the height of the edge dropped below *T1*.

Experimental Results

The source images are shown in Figs. 7.1, while the results are shown in Figs. 7.4, generated by MATLAB 6.1. We have tuned the parameters of the threshold and sigma such that the output is the most appropriate for comparison. The threshold (t) and sigma (δ) values used in Figs. 7.4 are: (a) $t = 0.5$ and $\delta = 0.5$; (b) $t = 0.5$ and $\delta = 0.6$; (c) $t = 0.5$ and $\delta = 0.9$; (d) $t = 0.4$ and $\delta = 1.0$.

The results are apparently better than that of Sobel and LoG edge detectors. However, it is still not good enough in the case of 'full of wrinkles'. In addition, Canny has the property of multi-resolution, i.e. by using different values of parameters such as sigma or threshold; we can get different details of line features from it. We must perform multiple passes on the image with different parameter values, to exploit all the different levels of lines features making it more complicated and hard to determine suitable parameters.



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

Fig. 7.4 Results obtained by Canny edge detector.

7.2 Thresholding Techniques

Apart from different edge detectors using gradient magnitude to find the edges, thresholding is another common approach which is able to separate the objects (edge) from the background, by analyzing the distribution of grayscale values of the image. If a pixel has an intensity value less than a predefined threshold value, the corresponding pixel in the resultant image is set to background (*0*), otherwise set to object (*1*). Generally, the performance of image thresholding methods are application dependent where no single method suitable for all kinds of images.

Thresholding methods can be categorized into two classes: global and local. In global thresholding method, one threshold value is chosen for the whole image [78-82]. In contrast, a threshold value is chosen for each small region or even in each pixel for

local adaptive thresholding method [83-91]. The global thresholding method is much efficient than local thresholding method since it has less computation involved. However, it is hard to define a good threshold value. Some global thresholding methods use the statistics on the histogram of the grayscale image to determine the threshold value, but it is not reliable since most of the images do not have clear peaks from the histogram. So, the adaptive thresholding methods are evolved to solve the problems from global thresholding methods.

7.2.1 Common Thresholding Methods

Otsu's thresholding method [92] is a well known method which is based on selecting the lowest point between two classes (peaks) of histogram. However, it uses an exhaustive search to evaluate the criteria for maximizing the between-class variance so that it takes a lot of computation time.

Chow and Kanenko [83] divide an image into an array of overlapping sub-images and then find the optimum threshold for each sub-image by investigating its histogram. The threshold for each single pixel is obtained by interpolating the results of the sub-images.

Niblack's method [84] proposed to threshold an image by calculating the mean and standard deviation. The threshold at pixel (x, y) is calculated by: $T(x, y) = m(x, y) + k.sd(x, y)$, where m is the mean value of pixels from a local window, $sd(x, y)$ is the standard deviation of the pixel values among the local window, and k is a constant used to adjust the performance of this algorithm.

7.2.2 Proposed Method – Adaptive Thresholding

The principal lines generally lie in a specific region on a palm, formed by flexing the hand and wrist in the palm, but those wrinkles go through all different directions. It is

hard to correctly extract the principal lines alone from a palm. Although our palmprint acquisition device is capable of obtaining the finest level of features – ridges, we found that not everybody has clear ridges on their palm. In view of this situation, our goal is only to get the major palm lines which are strong enough for the extraction. And most likely, these lines are the principal lines and wrinkles, but not ridges.

From the literature, Wu [93] uses the Canny edge detector to obtain the line features from a palm. Every line has two step-edges, and the width of the line is calculated by the distance between the two step-edges. The accuracy of their algorithm is dependent on the performance of the line features extracted. Their best recognition rate is 97.2%. We already evaluated the performance of some edge detectors and found that they are not satisfied on the palm lines extraction.

Here, we tried to use the adaptive thresholding method for the palm lines extraction. The main difference between adaptive thresholding method and global thresholding methods is former's thresholding value is not fixed. The threshold value keeps changing across the image, which is local to the image position. The rationale behind this is that we believe a small region in an image is more likely to have approximately uniform illumination, which gives a better result. There are different groups of people proposed their own adaptive thresholding method for image binarization. Our approach is borrowed from Niblack's method, by dividing an image into several overlapping regions, and then calculates the mean and standard deviation of the local region minus some adjustment constant to define the threshold value. We also added a false line removal process to detect regions with small grayscale values spread which was wrongly regarded as a line. The steps are described as follow:

Step 1: Window size selection

The window size $M \times N$ is an important parameter for this algorithm. We believe that a

smaller image region is more likely to have approximately uniform illumination. Too small window size makes the threshold susceptible to noise, but too large window size is violating the assumption of approximately uniform illumination. We must take care on the selection of this parameter by experiments.

Step 2: Sample mean $(\bar{x})$ ***and standard deviation*** (σ) ***calculation***

Calculate the sample mean $(\bar{x})$ and standard deviation (σ) of the pixel intensity values on the region specified by the window size $M \times N$ from *step 1*.

Step 3: False line removal process

if $(P_{high} - P_{low}) > c$

proceed to next step for the thresholding value calculation,

else set B(x, y) to background,

where P_{high} and P_{low} is the highest and lowest intensity value from the local window, and c is a constant.

The purpose of this step is to detect some regions where the grayscale values spread out in only small ranges may wrongly regarded as a line. If the difference between the highest and lowest values is less than a given constant, the pixel is regarded as background.

Step 4: Adaptive Thresholding calculation

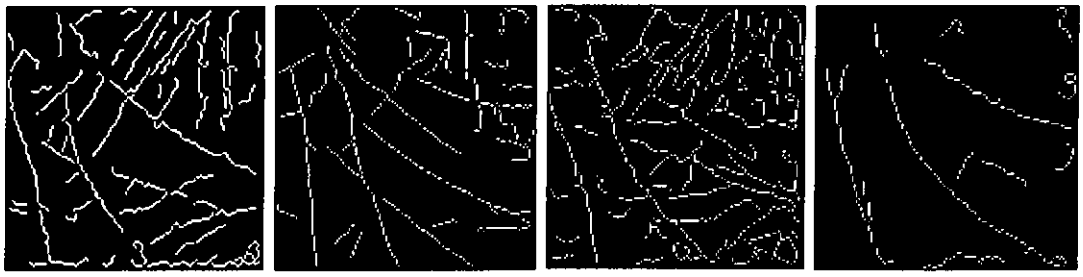
$$B(x,y) = \begin{cases} 0 & if \quad I(x,y) \le T(x,y) \\ 1 & if \quad I(x,y) > T(x,y) \end{cases}, \qquad (7\text{-}1)$$

where $T(x,y) = \bar{x} - k\sigma$; and k is a constant determined by experiments.

There are some width information on the binarized image map from the above steps. In order to get a better line representation of a palm, we performed a thinning process which can effectively outline the skeleton of the thick palm lines, into a one-pixel line representation. Some common thinning methods are reported in [94-95].

7.2.3 Experimental Results

The source images are shown in Figs. 7.1, while results are shown in Figs. 7.5. Our proposed method is implemented by MS Visual C++ 6.0. We have tuned the parameters of the *window size* (W), *constant* (c) to detect false line, and another *constant* (k) to adjust the performance, such that the output is the most appropriate for comparison. The parameters used in Figs. 7.5 are: (a) $W = 13$, $k = 7$ and $c = 15$; (b) $W = 13$, $k = 7$ and $c = 15$; (c) $W = 11$, $k = 11$ and $c = 15$; (d) $W = 13$, $k = 6$ and $c = 15$. Also, we have removed the lines with less than 15 pixels.



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

Fig. 7.5 Results obtained by Adaptive Thresholding.

It is apparent that even though the palmprint line features are very weak, we can still extract them appropriately. For those very strong principal lines and strong wrinkles, our method also performs well. By the way, this preliminary line map of the palmprint image can only provide an intermediate stage of result, as there are some artifacts such as broken lines, noise, etc on the map. It directs us to develop a knowledge based method to refine the result as revealed in the next Chapter.

7.3 Conclusions

Traditional edge detection techniques cannot extract the palm lines effectively because they are all dependent on the global thresholding technique which is not good for the peculiarity line features of a palm. Adaptive thresholding technique, on the other hand, uses a local window to calculate the thresholding value which is performed better than the global thresholding method. We first calculate the mean and standard deviation of the local region which then minus some adjustment constant to define the local threshold value. To avoid false line extraction, there is a checking process to detect regions with small grayscale values spreading which may in turn wrongly regarded as a line. From the experimental results, we found that Canny has the property of multi-resolution, i.e. by using different values of parameter such as sigma or threshold; we can get different details of line features from it. We must perform multiple passes on the image with different parameter values to exploit all the different levels of lines features. This makes the algorithm more complicated. Also, there must be some knowledge based rules on determining how to combine the output from different level of lines obtained.

For our proposed method, only few steps are needed to obtain all details of palm lines. It provides a solid foundation on the successive palm line features extraction. Next Chapter will discuss the line feature extraction incorporating knowledge of palm line structures.

8 LINE FEATURE EXTRACTION INCORPORATING KNOWLEDGE

We proposed to use a knowledge based method on the palm line extraction from the preliminary line map obtained by adaptive thresholding. We first defined the properties of line structures and line segments to create a palmprint structure map. Next, we designed the searching strategies to exploit the structural information of a palm. Some rules were established on solving the problems of 1) isolated points, 2) broken lines, 3) short bifurcation lines, 4) crossed over lines, and 5) looped lines. Finally, some line operations are also defined on joining or splitting a line so that they are more close to the original lines exists in the palm.

8.1 Create a Palmprint Structural Map

A palmprint structural map includes line structures and isolated points are shown in Figs. 8.1. We first classify each point into a category, according to its characteristic. Hilditch [96] defined the crossing number of a pixel, centered in a 9-grid window, $X_H(p)$, which is the number of times counting the cross over from a white point to a black point when the points in $N(p)$ are traversed in order, cutting the corner between 8-adjacent neighbors. There are four types of points from the preliminary line map can be obtained by the crossing number, which are:

1) Type 0 – isolation point, with crossing number equals to 0,

2) Type 1 – end point, with crossing number equals to 1,

3) Type 2 – intermediate point, with crossing number equals to 2, and

4) Type 3 – bifurcation point, with crossing numbers equal to 3/4.

Since both crossing numbers 3 and 4 exhibit the same properties – bifurcation, we put them to the same class, type 3.

A line structure is formed by one or more line segments. In Fig. 8.1 (a), there are two line structures and one isolated point, while Fig. 8.1 (b) uses the crossing numbers to describe the same line structures. The definitions of end point and bifurcation point are marked by blue and green circles respectively in the figure. These characteristics give the basic structures of palm line from the palmprint image. For example, there are six line segments on line structure 1 while there is only one line segment appeared in line structure 2.

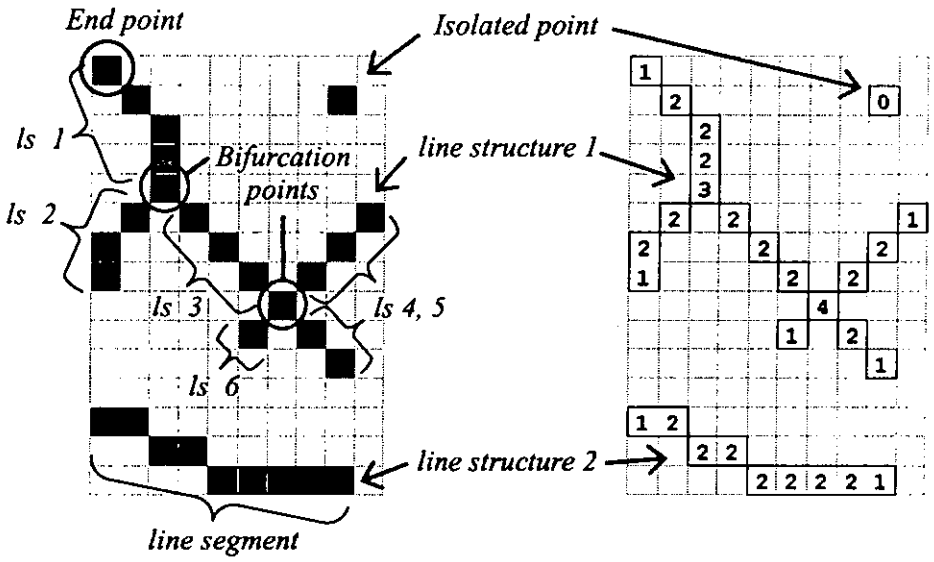


Fig. 8.1 Portion of palmprint structural map: (a) two line structures and an isolated point, (b) the crossing number of each pixel.

The major steps on creating the palmprint structural map are described as follow:

Step 1: Find all crossing numbers

Search the whole image pixel by pixel to get the type of each pixel.

Step 2: Point Classification

- For all those type 1 and type 3 points, they are the start point of searching.
- For all type 0 points, they are isolated points which are only useful when joining from broken lines.
- For all type 2 points, they are the intermediate points of a line, and guides on the searching.

Step 3: Trace from type 3 point (a bifurcation point)

The trace is started with a bifurcation point which goes through all its three/four branches, i.e. traced for 3/4 times. The trace follows the type 2 points until it meets a type 1 or 3 points. For example, there are two type 3 points in *line structure 1* from Fig. 8.1 (a). When the trace is started from the first type 3 point where its crossing number equals to 3, we can get the following line segments (*ls*):

1. 3, 2, 2, 2, 1 (*ls 1*)
2. 3, 2, 2, 1 (*ls 2*)
3. 3, 2, 2, 2, 4 (*ls 3*)

When the trace is started from the second type 3 point where its crossing number equals to 4, we can get the following line segments:

1. 4, 2, 2, 2, 3 (since this line segment is duplicated, we dropped this *line segment*)
2. 4, 2, 2, 1 (*ls 4*)
3. 4, 2, 1 (*ls 5*)
4. 4, 1 (*ls 6*)

The respective line segments are shown in Fig. 8.1 (a).

Step 4: Trace from type 1 point (an end point).

The trace follows the type 2 point until it meets a type 1 or 3 point. The only purpose of tracing from an end point is to get the type of line which are end point to end point, i.e. start with a type 1 point and ended with a type 1 point. This type of line structure is regarded as a short line or regular line depends on its length, and there is only one line segment in this type of structure. When the trace is started from a type 1 point, we can get the following line segments:

1. 1, 2, 2, 2, 3
2. 1, 2, 2, 3
3. 1, 4
4. 1, 2, 2, 4
5. 1, 2, 4

} Traced from *line structure 1*

6. 1, 2, 2, 2, 2, 2, 2, 2, 1
7. 1, 2, 2, 2, 2, 2, 2, 2, 1

} Traced from both ends of *line structure 2*

According to our purpose, we only take the line segments with type 1 point to type 1 point (i.e. to get the regular lines) , and ignore the remaining. So, we only consider the 6^{th} and 7^{th} lines from the above traces. Also, both 6^{th} and 7^{th} line segments are the same (by tracing from opposite direction), we only take one of them as the final line segment for the next stage processing.

After performing the above four steps, we can get all the line structures to form a complete palmprint structural map.

8.2 Properties of Line Segment

Every line segment from a line structure has three properties, including: 1) length, 2) direction and 3) type, described as follow:

1) Length of a line segment

It is important to know the length of a line segment since we can use this 'knowledge' to determine whether the line segment can be ignored, joined with another line segment, or

leave it as is. For example, if a line segment grows from a bifurcation point which is shorter than a predefined length, i.e. 3 pixels, we can disregard and remove it from the palmprint structural map. On the other hand, if it is longer than a predefined length, i.e. 16 pixels, we treat it as a normal line, etc.

The length can be calculated by:

$$Length = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}\ , \qquad (8\text{-}1)$$

where (x_1, y_1) and (x_2, y_2) are the start point and end point of a line segment, respectively.

2) Direction of a line segment

The direction of a particular line segment indicates the relationship between different line segments originated from a bifurcation point. From the literature of fingerprint, there are different methods proposed to estimate the orientation field of a fingerprint image, see [97-99]. A smooth orientation field fingerprint image can be estimated by its ridge structures using Rao's algorithm [99]. In our study, however, there is no need to perform those complicated calculations to get the orientation field of line segments from the palmprint image. The purpose on getting the direction of line segments is to determine the line operations, such as join or split from a bifurcation point, if necessary. The direction of a line segment is calculated by two points, one is the point of bifurcation (x_2, y_2), while the other point, (x_1, y_1), is defined by tracing points from the bifurcation point with a predefined distance, i.e. 12 pixels. If the length of the line segment is less than the predefined distance, then (x_1, y_1) is the position of the last pixel. We have defined 8 directions, from 0 to 7, which is calculated by the quantization of an angle between the line segments:

$$Direction = abs(\theta / 22.5)\ , \qquad (8\text{-}2)$$

where

$$Angle\ (\theta) = \tan^{-1}\frac{(x_2 - x_1)}{(y_2 - y_1)}, \qquad (8\text{-}3)$$

where (x_1, y_1) and (x_2, y_2) are the end points of a line segment.

3) Type of a line segment

There are three types of line segments from the palm line structures, they are:

i. Type 1: bifurcation-to-bifurcation (b-b)

ii. Type 2: bifurcation-to-ending (b-e)

iii. Type 3: ending-to-ending (e-e)

Having classified the type of points on the preliminary line map of a palmprint image, we can further define the type of line by looking at the type of its ends. For example, line segments (*ls*) 1, 2, 4, 5, and 6 from *line structure 1* in Fig. 8.1 (a) are regarded as a type 2 line segment, while *ls* 3 is regarded as a type 1 line segment. Finally, the regular line segment in *line structure 2* is regarded as a type 3 line segment.

8.3 Searching Strategies

8.3.1 Searching from a Bifurcation Point

The search starting from a bifurcation point needs to run *n* times where *n* equal to the number of branches it has. Here we use the term '*branch*' to describe the line segment traced from a bifurcation point(s) since branch is more appropriate word in this context. Four typical cases of problems can be exploited by this search, they are: 1) single bifurcation, 2) complex bifurcation – looping, 3) complex bifurcation – crossover, and 4) complex bifurcation – others. There is a split operation performed on this search in order to remove the 'bifurcation point' (marked in blue color as seen from Figs. 8.2) and the first pixel of each branch started from that bifurcation point (marked in red color as seen from Figs. 8.2) such that all the line segments are broken down. The main purposes of this split operation are to make the messy lines separated and to remove some possible

wrong lines formed. They may be joined together or removed at later stage if they meet the joining or removing criteria.

Case 1: Single bifurcation

There is only one bifurcation point in a set of connected line segments, as shown in Figs. 8.2 (a1 and a2). The resulting line structure is shown in Fig. 8.2 (a3) with some points removed.

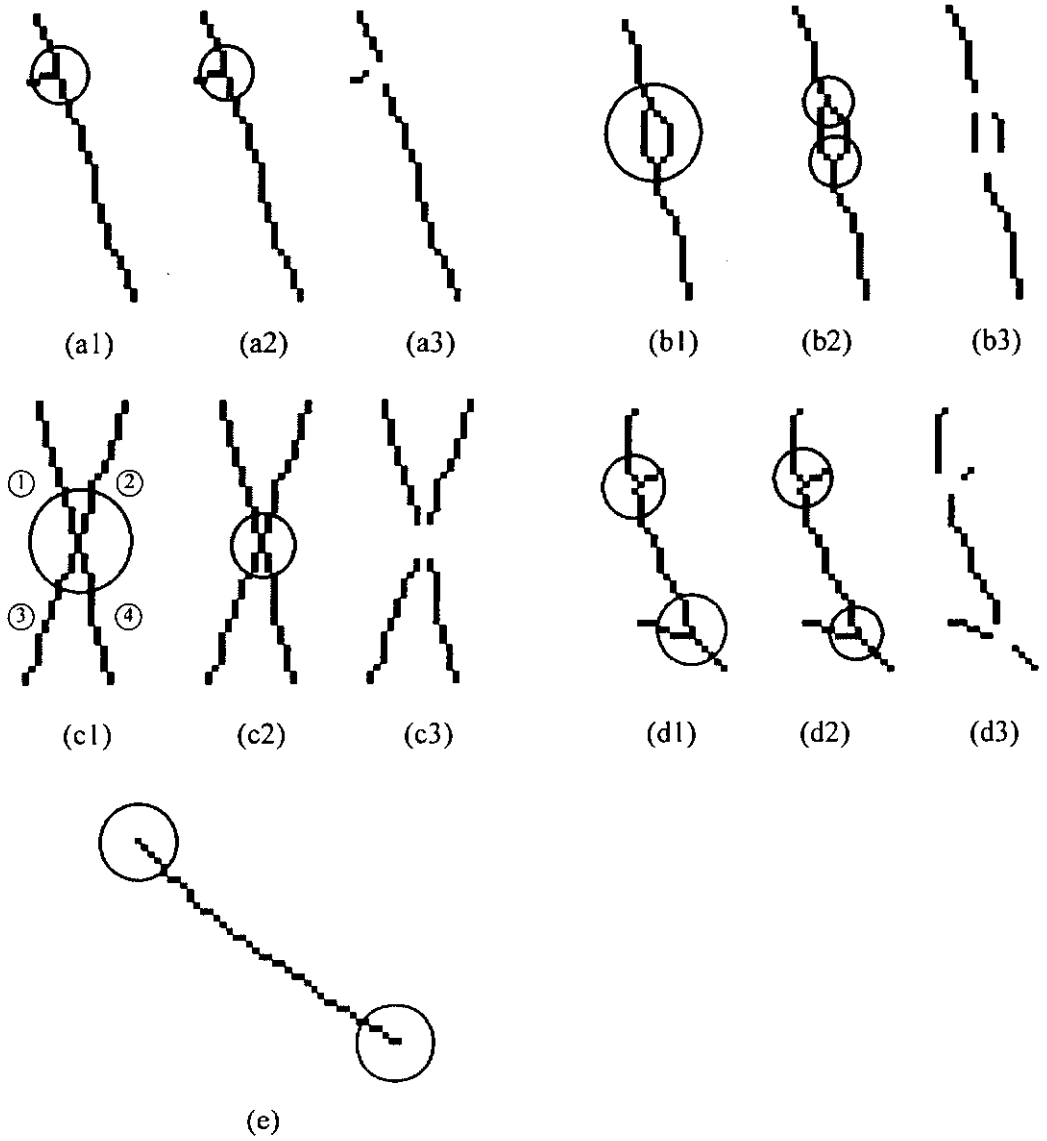


Fig. 8.2 Five typical line structures found in a preliminary line map of a palmprint image.

Case 2: Complex bifurcation - Looping

It is formed when two bifurcation points are closed together in parallel to form a looping line structure, as shown in Figs. 8.2 (b1 and b2). The resulting line structure is shown in Fig. 8.2 (b3) with some points removed.

Case 3: Complex bifurcation – Crossover

This case is formed by one or more bifurcation points appeared together to form a crossed line structure, as shown in Figs. 8.2 (c1 and c2). The resulting line structure is shown in Fig. 8.2 (c3) with some points removed. This scenario is a more complex one and more knowledge are required for the line operations. In-depth discussion is found on later Section.

Case 4: Complex bifurcation – Others

This case is formed by more than one bifurcation points appeared together to form a complex line structure, but it is not one of the above cases. Here we take an example of two bifurcations points with one common branch in a line structure, as shown in Figs. 8.2 (d1 and d2). The resulting line structure is shown in Fig. 8.2 (d3) with some points removed.

8.3.2 Searching from an End Point

This searching strategy is purposed on exploiting the regular lines found in the preliminary line map of a palmprint image. We only take care on type 3 branches, i.e. ending-to-ending.

Case 5: Regular line

There is only one line segment appeared in a line structure for regular line, i.e. no bifurcation point appeared, as shown in Fig. 8.2 (e). If the length of this type 3 line is less then a given threshold, i.e. 3 pixels, then it is removed from the palmprint structural map.

8.4 Line Operations

Having defined the properties of line segment and searching strategies, we can define the line operations and the rules governing those operations target on solving the five major problems, i.e. isolated points, broken lines, short bifurcation lines, crossed over lines, and looped lines.

8.4.1 Handling Isolated Points

This scenario is produced by the noise of the palmprint image. It is identified by a point with crossing number equals to zero, i.e. type 0 point, as shown in Fig. 8.1 (a). We can solve this problem by deleting all type 0 points from the map directly.

8.4.2 Handling Broken Lines

Most of time, there are noises appeared on the binary thinned image so that a line is easily broken into a few shorter line segments. The following steps are defined for the purpose of connecting broken short line segments into a single, complete line.

Step 1: Find a set of candidate(s)

If there are two or more lines which are opposite to each other, with a distance less than a given threshold value, i.e. 6 pixels, then we can put these lines in a candidate set for the calculations in *steps 2 to 7*.

Step 2: Calculate angles (θ_1, θ_2) of the lines concerned.

First take two lines (line 1 and line 2) from the set of candidates, with end points (i_1, j_1) and (i_2, j_2) respectively. Then, the angles θ_1 and θ_2 of these two lines can be calculated from the position starting from that end point with n^{th} pixels tracing from it, i.e. 16 pixels, which are (x_1, y_1) and (x_2, y_2) respectively, as shown in Fig. 8.3.

$$\theta_1 = \tan^{-1}\frac{(j_1 - y_1)}{(i_1 - x_1)}, \qquad (8\text{-}4)$$

where (i_1, j_1) and (x_1, y_1) are the end points of line 1.

$$\theta_2 = \tan^{-1}\frac{(y_2 - j_2)}{(x_2 - i_2)}, \qquad (8\text{-}5)$$

where (i_2, j_2) and (x_2, y_2) are the end points of line 2.

Step 3: Calculate angle (θ_3) between the lines concerned.

The direction of a line starting from (i_1, j_1) and ended at (i_2, j_2) is defined as θ_3, as shown in Fig. 8.3. The calculation is similar to equation *(8-5)*.

Step 4: Determine the angle difference between the two lines

The absolute difference of two angles, θ_d, from two lines should be close to 180°.

$$\theta_d = 180° - |\theta_2 - \theta_1| \qquad (8\text{-}6)$$

If the resulting angle is within the range of $\pm 60^\circ$, proceed to next step.

Step 5: Determine the smooth flow of the joining line

Similarly, both the angle differences between $(\theta_3 - \theta_1)$ and $(\theta_2 - \theta_3)$ should be less than $\pm 60^\circ$, so that the direction of a line can only be changed smoothly,

Step 6: Determine the distance between two lines

The distance between the two endings must be less than a given value, i.e. 6 pixels.

Step 7: Choose the best segment to join.

Every line has its start point and end point. We calculate the best matching line it can join to, if there is one, according to the above criteria. We only perform joining operation *if and only if* both lines have best matching line from each other. For example, from Table 8.1, we found that both line 1's end and line 2's end are best matched with line 3's start, but the best matching line from line 3's start is pointed to line 2's end. That means the join operation can only performed on line 2's end with line 3's start.

Table 8.1 Line joining table.

Line number	Start/End Point	Best Matching
1	Start	N/A
1	End	3 Start
2	Start	1 End
2	**End**	**3 Start**
3	**Start**	**2 End**
3	End	N/A

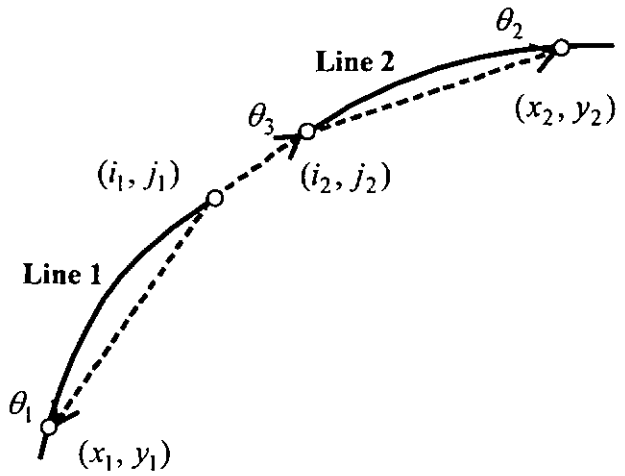
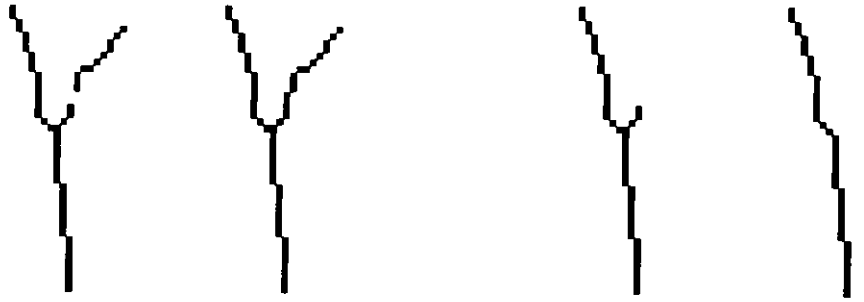


Fig. 8.3 Illustration of the points and angles involved on broken line structures.

8.4.3 Handling Short Bifurcations

Short bifurcations are formed by two possible situations: 1) noise occurred to break down the regular bifurcation segments, i.e. it should be a regular bifurcation segment with 3 long branches. 2) Noise occurred to add a 'wrong' branch to form the bifurcation point. As revealed from the searching strategy on *single bifurcation*, we have removed bifurcation point to form three separate line segments. For the first problem as shown in Fig. 8.4 (a1), there is another short line near the short bifurcation line, and finally, they

joined together by the "*broken line strategy*" routines as described in Section 8.4.2 since the criteria of join operation have met, as shown in Fig. 8.4 (a2). For the second problem shown in Fig. 8.4 (b1), there is no line nearby the short line created by the bifurcation, so we removed this short bifurcation branch to obtain a smooth line, as shown in Fig. 8.4 (b2).



(a1) (a2) (b1) (b2)

Fig. 8.4 Two cases on bifurcation: (a) a short line nearby, and (b) no short line nearby.

8.4.4 Handling Crossed Over Lines

Crossed over lines are formed by one or more bifurcation points together appeared in the same line structure. We first trace from the bifurcation point, and perform the "*broken line strategy*" routine as described in Section 8.4.2. There are at least four possible line structures formed as follow, from *ls* ①, ②, ③, and ④, as referring to Fig. 8.2 (c1):

1. *ls* ① joined with *ls* ④ while *ls* ② joined with *ls* ③,
2. *ls* ① joined with *ls* ③ while *ls* ② joined with *ls* ④,
3. *ls* ① joined with *ls* ④ while *ls* ② and *ls* ③ are separated lines which cannot be joined together, and
4. *ls* ② joined with *ls* ③ while *ls* ① and *ls* ④ are separated lines which cannot be joined together.

Any cases of join/non-join operations are totally determined by the rules defined and knowledge obtained from the angles involved between the line segments and the properties of the line segments involved.

8.4.5 Handling Looped Lines

Loop is formed when two bifurcations appeared close together in a line structure, but there is different possible physical layout. Basically, there are four line segments: *ls* ①, ②, ③, and ④ appeared to form a loop in the line structure with three types of layouts as shown in Figs. 8.5, they are:

1. left and right branch symmetry, as shown in Fig. 8.5 (a1),
2. left branch is a large curved line while right branch is nearly a straight line, as shown in Fig. 8.5 (b1), and
3. flipped of the second layout, as shown in Fig. 8.5 (c1).

Actually, when we found that there are bifurcations at a line structure, we removed the bifurcation point and its immediate point on each branches, as mentioned in Section 8.3.1. Then we trace from each bifurcation point, and perform the "*broken line strategy*" routine as described in Section 8.4.2. For layouts 2 and 3 from Figs. 8.5 (b1 and c1), a smooth line can be formed easily by joining the upper *ls* ① to bottom *ls* ④ through *ls* ③ (left loop), or through *ls* ② (right loop), respectively, i.e. where the best matching line is formed. That is, the branch marked in red color in Figs. 8.5 (b2 and c2) are voided so that the looped line structure are cleared and the resulting line structures are shown in Figs. 8.5 (b3 and c3). On the other hand, in case of the first layout from Fig. 8.5 (a1) where the best matching line may be not formed since both sides may get similar significance, i.e. the best matching line of *ls* ①'s end is *ls* ③'s start while the best matching line of *ls* ②'s end is *ls* ④'s start, so that no line segment can be joined together according to the

rules. So we adopted an additional routine to solve this problem. We need to look forward for one more line to figure out the best matching line, if any. In fact, *ls* ①'s end and *ls* ④'s start are the best matching using this additional rule. We can join them together by a blue line, and voided the branches ② and ③ marked in red color, as shown in Fig. 8.5 (a2) and the resulted line segment is shown in Fig. 8.5 (a3).

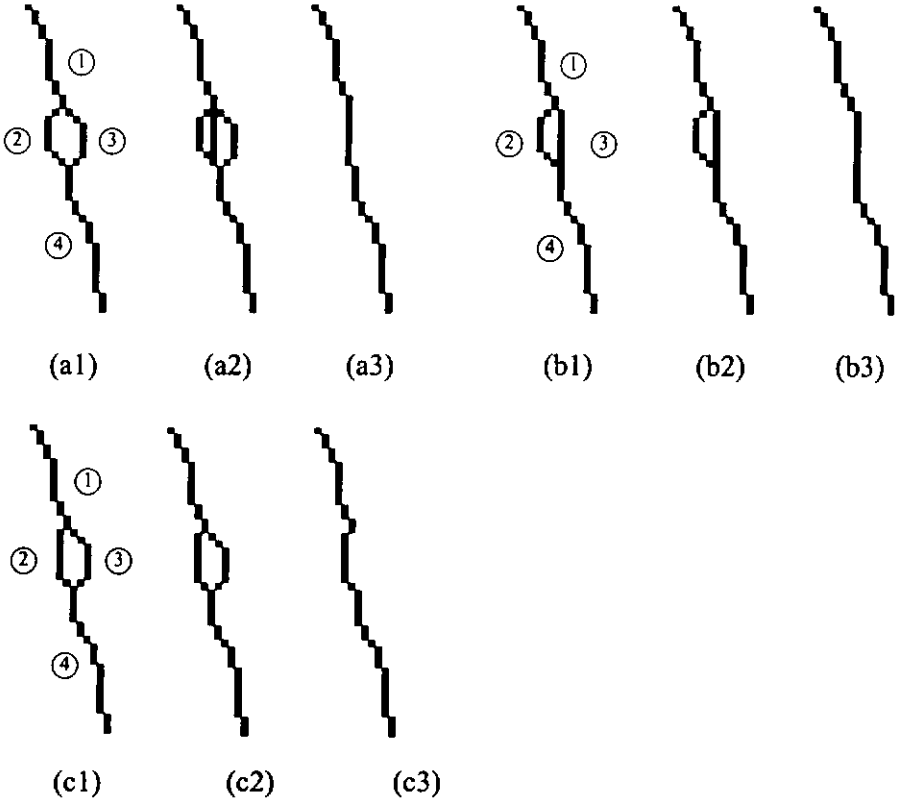


Fig. 8.5 Three typical layouts of a looped line structure.

8.5 Line Curvature Checking

When the lines were extracted from the palmprint structural map using prior knowledge, the lines formed may be not close to the original palm lines. According to our experiences, the direction of a line is usually changed smoothly, i.e. without abrupt changes in direction. In view of this characteristic, we should add a procedure to check whether the lines obtained by our procedures are conformed to the natural characteristics of palm

lines. Fig. 8.6 (a) shows a line with acceptable curvature while Fig. 8.6 (b) shows a line with exceptional large curvature with large D and big angle created.

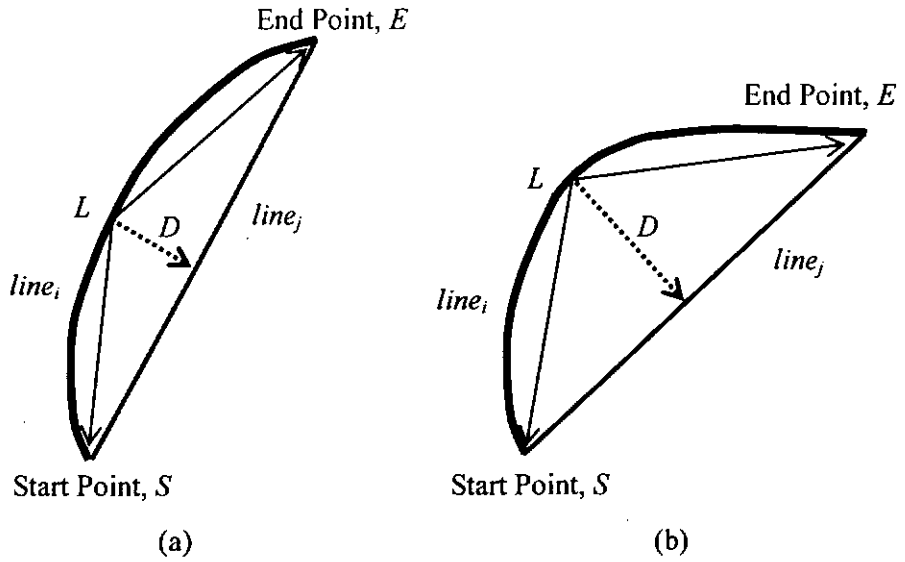


Fig. 8.6 (a) a normal curved line, and (b) a large curved line.

Let the line to be checked is called $line_i$, the following are steps of line checking:

Step 1: Find the point of maximum distance

Make a straight line joining the start point (S) and end point (E) of $line_i$ called $line_j$, then, find a point (L) from $line_i$ which has the longest distance D and is perpendicular to $line_j$.

Step 2: Check the angle involved

If distance D is longer than a predefined threshold, then we need to check the angle $\angle SLE$ at point L from $line_i$ to see whether it is close to $180°$, i.e. a smooth line.

$$angle\ (\theta) = 180 - \angle SLE \qquad (8\text{-}7)$$

If $angle\ (\theta)$ is larger than the threshold, then we need to further check the line according to the pseudo code listed in *Step 3* to see whether we need to split the line.

Step 3: Pseudo Code for the Line Curvature Checking

```
do {
    if line_length > 20
        if angle (θ) > 40
            set the current point (L) as end point
            check again until we found the best splitting point
            set splitting to true
        else
            if (splitting != true) then exit
            split the line at the current point position
            set the start point to current point position
            set end point equal to End Point (E)
            check the remaining points to see whether we need to split
            set splitting to false
    else
        if angle (θ) > 30
            set the current point (L) as end point
            check again until we found the best splitting point
            set splitting to true
        else
            if (splitting != true) then exit
            set the start point to current point position
            set end point equal to End Point (E)
            check the remaining points to see whether we need to split
            set splitting to false
} while checking not finished.
```

8.6 Sample Results

We collected a set of experiment results in order to demonstrate the performance of our proposed method based on the knowledge from the line structures. The samples illustrated are obtained from the preliminary line map of a palmprint image as described in Section 7.2.2 of Chapter 7.

There are five knowledge based method targeted on handling problems of ① isolated points, ② broken lines, ③ short bifurcation lines, ④ crossed over lines, and ⑤ looped lines, respectively. Figs. 8.7 and Figs. 8.8 list examples on those mentioned problems. All the palm lines shown in the sample images have minimum line length of 20 pixels. The leftmost column is the original central part sub-image of a palm. The preliminary line map of palmprint images are shown in the middle column, while the final images are shown in the rightmost column. There are pink colored circles with a number near it on identifying the type of problems were solved, i.e. ① points to the problem of isolated points, to show the effectiveness of our method.

From the sample results, we can say that using knowledge from the palm line structures can effectively enhance the performance on palm line extraction, compared with the preliminary line map of a palmprint image at the middle column.

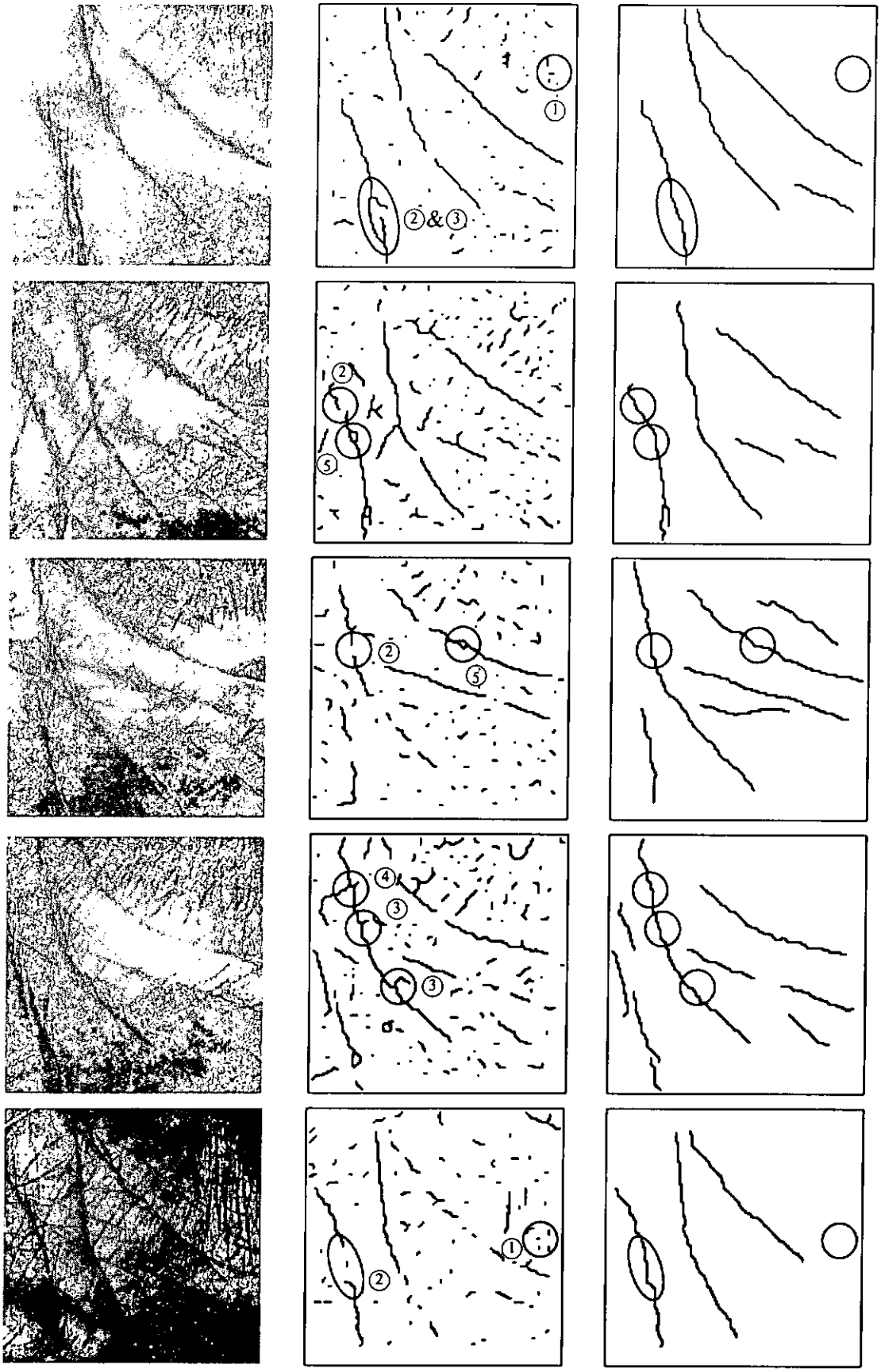


Fig. 8.7 From left to right: original central part sub-image of a palm, preliminary line map of palmprint images and final images.

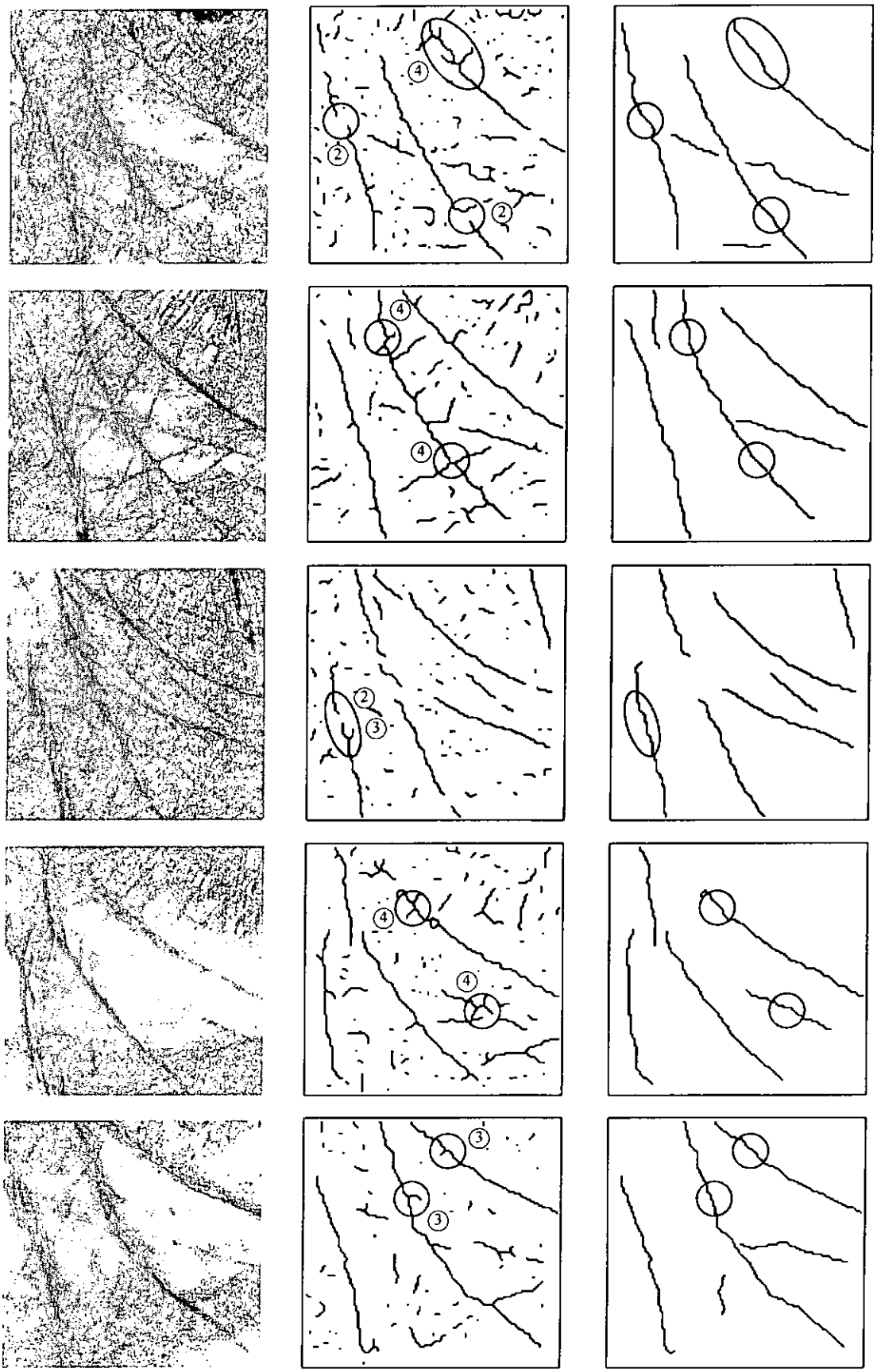


Fig. 8.8 More sample results, from left to right: original central part sub-image of a palm, preliminary line map of a palmprint images and final images.

8.7 Conclusions

There are five major problems from the preliminary line map of a palmprint image, they are: 1) isolated points, 2) broken lines, 3) short bifurcation lines, 4) crossed over lines, and 5) looped lines. In this Chapter, we proposed a method to promote the accuracy of palm line extraction by incorporating knowledge. We first defined the bifurcation points and end points in a palmprint image to create sets of line segments and structures, and finally formed the complete palmprint structural map. Next, we defined the properties and type of line segment. Then, we also defined searching strategies to exploit the whole structural information of a palm, they are: 1) single bifurcation, 2) complex bifurcation – looping, 3) complex bifurcation – crossover, 4) complex bifurcation – other cases, and 5) single line. We defined some rules on line operations for joining or splitting a line so as to correct the problems occurred. Different problems are solved, such as: 1) isolated points, 2) broken lines, 3) short bifurcation lines, 4) crossed over lines, and 5) looped lines. As shown from the images on the sample results, all the problems mentioned are solved. We can conclude that by incorporating the above rules with the knowledge from the palm line structures, the major palm lines can be extracted effectively.

9 MATCHING OF PALM LINES

In line matching stage, the identification template is compared with the master templates such that the system determines whether they are come from the same person. There are different methods on matching two set of features, but the selections are highly dependent on the type of features and their representations. Some matching strategies on palmprint are proposed by different researchers [31, 45, 48, 50-51]. We have defined rules on tracing lines from the palmprint image as discussed in Chapter 8. All lines are stored as a set of points in sequence. Our strategies on matching are designed by first defining a bounding box to limit the searching area. Next, we divide each line into shorter line called nodes, and an angle comparison is performed node by node to give the similarity scores. The higher scores of two palms mean the more similar they are.

9.1 Matching Strategies

Matching of palm lines is to find the differences between two palms, i and j. There are four major steps on the matching strategies, they are:

Step 1: Define a bounding box

We first defined the points (x_1, y_1) and (x_2, y_2), which is the start point and end point of *palm i*, respectively. Then, we use (x_1, y_1) as the upper left corner and (x_2, y_2) as lower right corner to make a rectangle box called "bounding box". Fig. 9.1 (a) shows the line from *palm i* and Fig. 9.1 (b) shows the line from *palm j*, with the bounding box

marked by dotted lines. In order to take care on the translation problem, we will extend the area of the bounding box from (x_1, y_1) to (ex_1, ey_1) and (x_2, y_2) to (ex_2, ey_2) respectively, by adding a few pixels to each side (i.e. here we use 3 pixels), as shown in Fig. 9.1 (b) marked with the outer box.

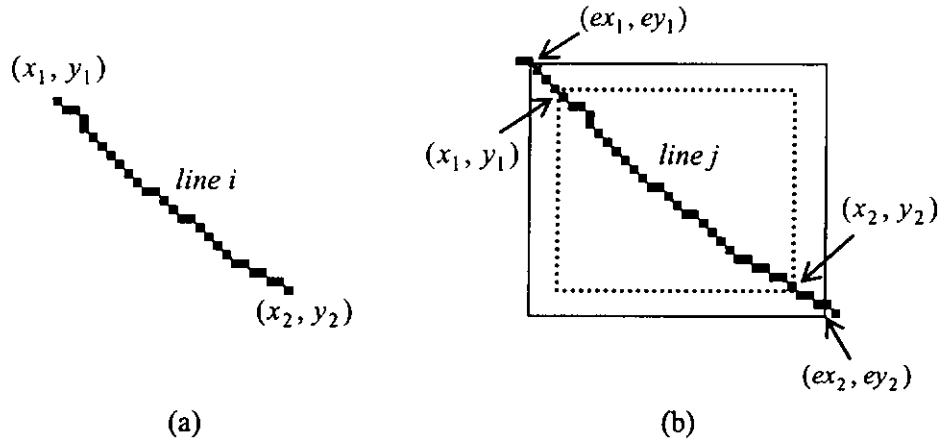


Fig. 9.1 Bounding box definition.

Step 2: Select a line(s) from palm j for matching

We calculate the number of points that each line from *palm j* was lay inside this bounding box. If the length of the second longest line is very close to the longest line, i.e. their difference is less than 10 pixels, we take both lines to the candidate set for the matching, and use the higher score's line for the matching score.

Step 3: Split the line into nodes

We split the line into a set of nodes, where the node size is set to 6 (points) here, as shown in Fig. 9.2. If the points in a line are not divisible by the node size (i.e. 6), and the remaining points are less than half of the node size, the remaining points are disregarded, else we use the remaining points to form the last node. Each node forms a short line with angle of inclination by joining its start point and end point.

Step 4: Assign score

We compare the angle of each node from *line i* to the angle of the corresponding node in *line j*. The score assigned depends on the similarity of the angles involved.

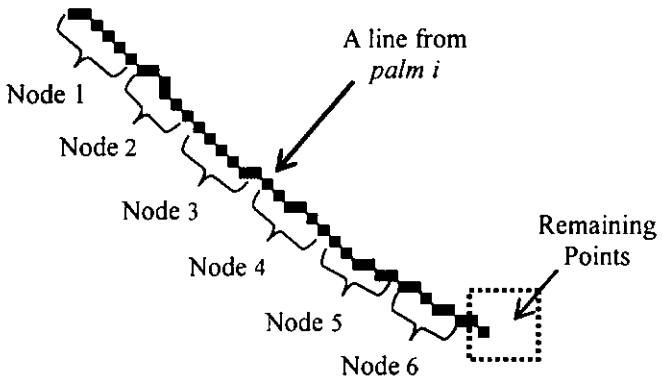


Fig. 9.2 A line is converted to a set of nodes with node size equals to 6.

9.2 Matching Score Calculations

According to the matching strategies defined, we can perform the matching score calculations by comparing the angles of the nodes involved, and assigning a matching score to the total score.

$$Score = \frac{\sum_{m=1}^{maxLineOf\_l} L(l_m, cl)}{n}, \qquad (9\text{-}1)$$

where l is a line from *palm i*, cl is the corresponding line from the bounding box, n is the number of comparison, and

$$L(l, cl) = \sum_{i=1}^{maxNodeOf\_l} NS(l\_Node_i, cl\_Node_i), \qquad (9\text{-}2)$$

where *l_Node* is a node from the line l and *cl_Node* is a node from the line cl, and

$$NS(i, j) = 1 \text{ if } diff(i, j) < T1;$$
$$\text{else } NS(i, j) = 0.7 \text{ if } diff(i, j) < T2;$$
$$\text{else } NS(i, j) = 0; \qquad (9\text{-}3)$$

where *diff(i, j)* is the angle difference between *node i* and *node j*, and $T1$ and $T2$ are two thresholds controlling the score assigned to a node with different levels of similarity. Here we set $T1 = \pm 7°$ and $T2 = \pm 13°$

If the nodes of cl is less than nodes from l (i.e. cl is shorter than l), or there is no

line appeared in the bounding box, i.e. *cl* is *null*, then, the number of comparisons (n) is still accumulated (i.e. adding the number of nodes from the line l), but the matching score is zero for all null nodes here.

If there are two lines from the corresponding bounding box need to be matched, then we match them one by one and take the maximum score from them.

$$Score = \frac{\sum_{m=1}^{maxLine} \max(L(l_m, cl_1), L(l_m, cl_2))}{n}, \quad (9\text{-}4)$$

where l is a line from *palm i*, cl_1 and cl_2 are the corresponding lines from the bounding box, and n is the number of comparisons.

9.3 Parameter Selection Considering Rotation and Shifting

It is not unusual to have a palmprint image rotated and/or translated a little bit. Our matching scheme has implemented some forms of measurement to solve this situation.

9.3.1 Shift and Match

In order to solve the line shifting problems, we shift the points and nodes from *palm j* in order to get better results. There are two types of shifting: 1) point shifting, and 2) node shifting.

1. Point Shifting

According to the node creation procedures described in Section 9.1, the first node is formed from the first point of a line involved. By introducing the point shifting idea, it means that the first point of the first node is located at point n of the line, where n is the number of point shifted and is within the range of *0* to *nodeSize-1*. Since the node size is relative short (i.e. 6 points) and the angle change is not much, there is no need to perform shifting point by point for 6 times, i.e. maximum *nodeSize*. We only performed one normal operation and one shifted operation in which 3 points are shifted (i.e. half of the

node size), and finally took the maximum score from them.

$$Score = \frac{\sum_{m=1}^{maxLineOf\_l} \max_{p=0,3} L(l_m, cl^p)}{n}, \quad (9\text{-}5)$$

where l is a line from *palm i*, cl is the corresponding line from the bounding box, p is the number of points shifted, and n is the number of comparisons.

2. Node Shifting

Similarly, node shifting is implemented to solve the translation problem. Suppose there are two lines (l *and* cl) to match, we performed one normal and one shifted matching:

$$Score = \frac{\sum_{m=1}^{maxLineOf\_l} \max_{nn=0,1} L(l_m, cl^{nn})}{n}, \quad (9\text{-}6)$$

where l is a line from *palm i*, cl is the corresponding line from the bounding box, nn is the number of node shifted, and n is the number of comparisons.

9.3.2 Bidirectional Matching

In order to enhance the accuracy of palm line matching, we use a bidirectional matching scheme which match *palm i* to *palm j*, and then reverse it by matching *palm j* to *palm i*. The matching score is the mean of these two operations.

$$Bidirectional\ Matching\ Score = \frac{Score_{i\_j} + Score_{j\_i}}{2}, \quad (9\text{-}7)$$

where $Score_{i\_j}$ is a score obtained by matching *palm i* to *palm j* while $Score_{j\_i}$ is a score obtained by matching *palm j* to *palm i*.

9.4 Experimental Results

The experiments are based on the central part sub-image which was obtained using the method from Section 6.1.4 of Chapter 6. We applied the adaptive thresholding technique to get the preliminary line map of a palmprint image as described from Chapter 7 Section

7.2.2. Next, we performed a set of operation on the line map incorporating knowledge, to enhance the accuracy of the line extraction, as described in Chapter 8. In order to examine the effectiveness of the knowledge based palm line extraction, we performed a lot of experiments with different parameters setting so as to choose the best parameters. The implementation program for the experiments is written by MS Visual C++ 6.0.

We use the same database as mentioned in Section 5.5 of Chapter 5. We randomly take out 150 persons from the database, with 5 images per person to conduct the experimental testing. Based on the statistics obtained, we can found out the weaknesses of our method and make enhancement in the future. Actually, there are few parameters/matching strategies may affect the performance, so our experiment settings are focused on finding out the best parameters and strategies on matching.

Parameters Setting

1. *n*, the *node size* equals to 5 to 8
2. *w1*, the *weighting of angles* > 7° and < 13° , assign score ranging from 0.6 to 0.8
3. *w2*, the *weighting of shifted points/nodes*, assign score ranging from 0.6 to 0.8

Matching Strategies

1. *m1*, using the score of *palm i* matching with *palm j* only.
2. *m2*, using the score of *palm j* matching with *palm i* only.
3. *m3*, using the higher score of 1 and 2 above, i.e. *Max(m1, m2)*.
4. *m4*, using the averaged score of 1 and 2 above, i.e. *Mean(m1, m2)*.

We summarized the results of the best parameters of *w1* and *w2* obtained using *n* equals to 5-8 in Table 9.1. Equal Error Rate (EER) is computed as the point where False Rejection Rate (FRR) = False Acceptance Rate (FAR). Since the matching score distributions are not continuous and a crossover point might not exist, we use the method proposed by Maio et al. [100] to calculate the EER:

$$[EER_{low}, EER_{high}] = \begin{cases} [FRR(t1), FAR(t1)] \; if \; FRR(t1) + FAR(t1) \le FRR(t2) + FAR(t2) \\ [FAR(t2), FRR(t2)] \; otherwise \end{cases}, \quad (9\text{-}8)$$

and then EER is estimated by $(EER_{low} + EER_{high})/2$.

We found that the matching strategy *m4*, *Mean(m1, m2)*, provides the best result for different node size. From the table, it can be seen that the best parameter combinations are: $n = 7$, $w1 = 0.6$ and $w2 = 0.7$ which produce $EER = 17.6\%$ at $Threshold = 44$.

Table 9.1 The best results for $n = 5$ to 8, with the best parameters of *w1* and *w2*

Node Size	*Parameters*	*m1*	*m2*	*m3*	*m4*
5	*w1*	*0.5*	*0.5*	*0.5*	*0.5*
	w2	*0.5*	*0.8*	*0.8*	*0.8*
	Threshold (0-100)	*41*	*43*	*46*	*42*
	EER (%)	*21.8*	*21.4*	*20.9*	*18.3*
6	*w1*	*0.5*	*0.5*	*0.6*	*0.5*
	w2	*0.6*	*0.6*	*0.6*	*0.5*
	Threshold (0-100)	*42*	*42*	*47*	*41*
	EER (%)	*20.9*	*20.9*	*20.6*	*17.8*
7	*w1*	*0.6*	*0.6*	*0.6*	*0.6*
	w2	*0.7*	*0.7*	*0.8*	*0.7*
	Threshold (0-100)	*44*	*44*	*49*	*44*
	EER (%)	*20.6*	*20.6*	*19.9*	*17.6*
8	*w1*	*0.5*	*0.5*	*0.5*	*0.5*
	w2	*0.6*	*0.6*	*0.6*	*0.7*
	Threshold (0-100)	*45*	*45*	*50*	*45*
	EER (%)	*20.6*	*20.8*	*19.4*	*18.0*

Fig. 9.3 shows the curves of FRR and FAR with EER formed at 17.6% and threshold at 44. The same results are plotted as ROC curve as shown in Fig. 9.4. We obtained the execution time on different stages of the line based recognition engine including preprocessing (the binarization and thinning of the image, from the central part sub-image), feature extraction and matching, as shown in Table 9.2. The algorithm is implemented using Visual C++ 6.0 on a PC with Intel Pentium 4 processor 2GHz and 1GB RAM. The total execution time is only 471 *ms*.

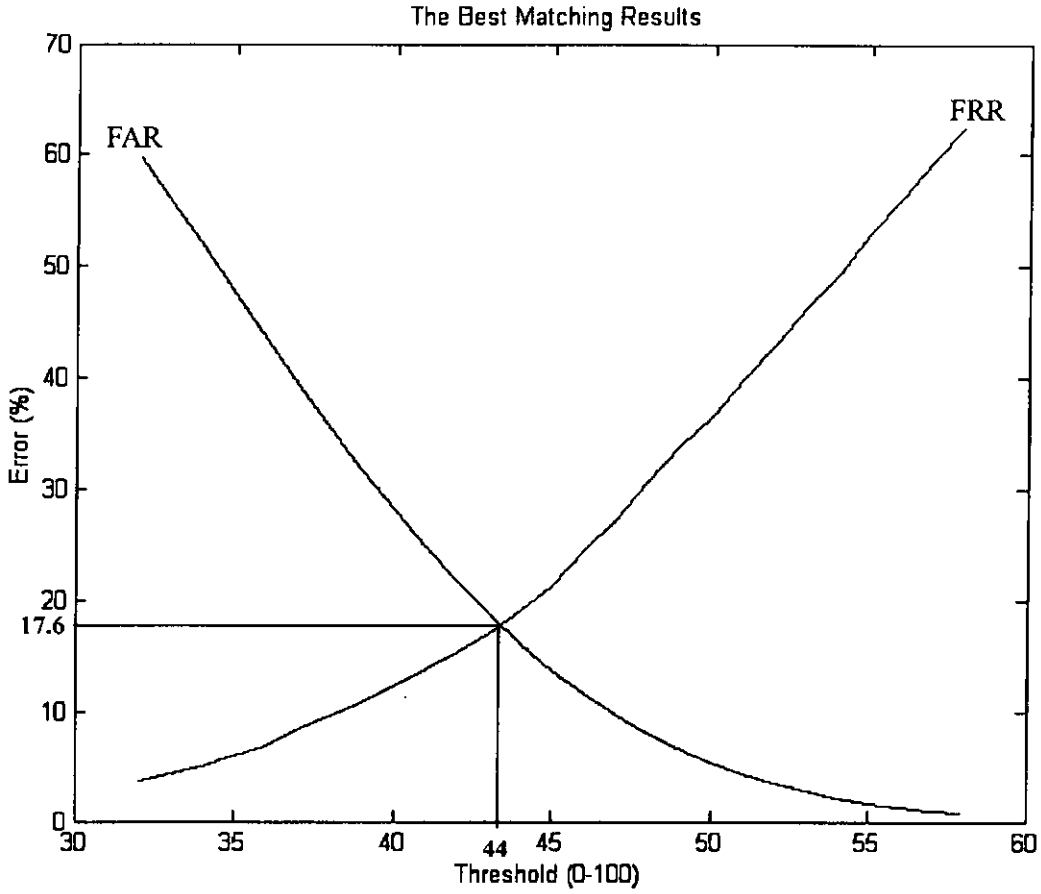


Fig. 9.3 The curves of FRR and FAR with EER formed at 17.6% and threshold at 44.

Table 9.2 Execution time on different stages of the line based recognition engine.

Operation	**Execution Time**
Preprocessing – binarization and thinning	97 *ms*
Feature Extraction	374 *ms*
Matching	0.48 *ms*
Total	471.48 *ms*

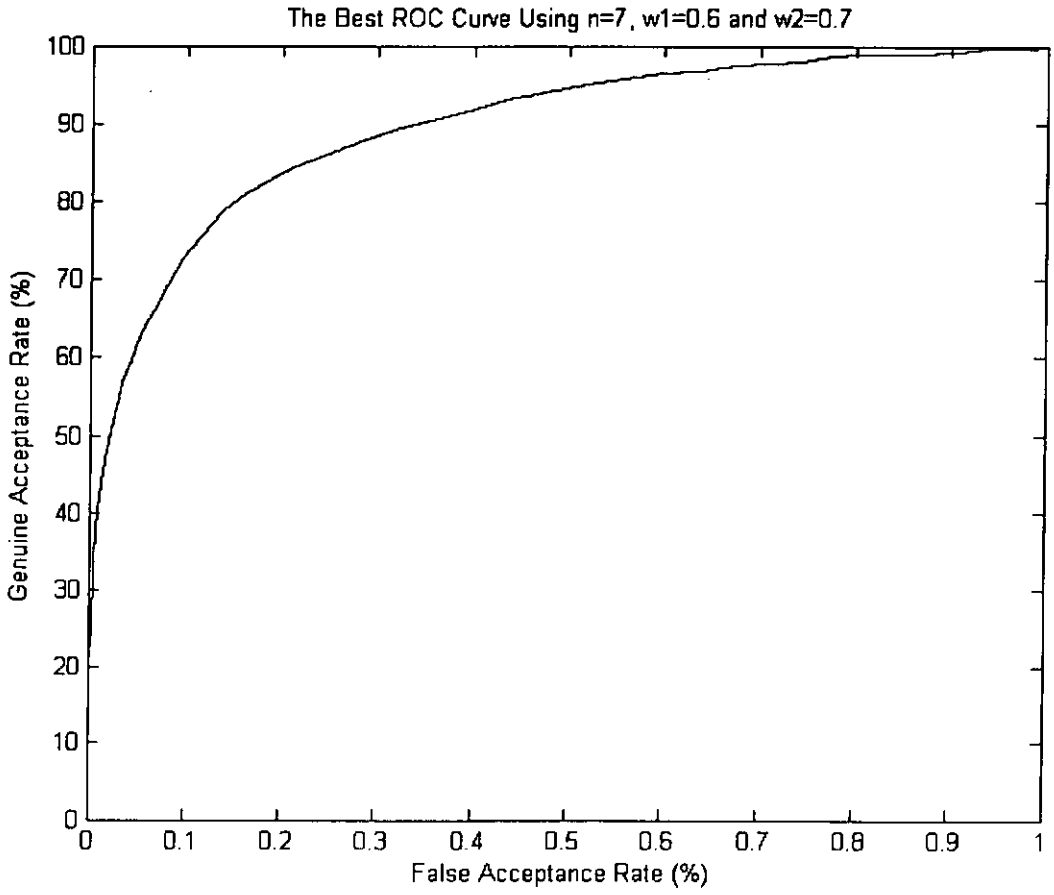


Fig. 9.4 The ROC curve of the best results.

9.5 Conclusions

From the experiments results obtained, the algorithm only achieves the EER at 17.6% (Recognition rate at 82.4%) for the best parameters. We passed through different stages of operations such as: the binarization of the palmprint image using adaptive thresholding, rules definition for line operations, matching strategies, and parameter settings. We think that we can refine the parameters/rules on different stages so as to obtain a better result in the near future, i.e. EER at < 10%. Although this recognition rate is still not satisfied for a personal identification system, we think that the line based approach can make contributions on the classification of palmprint, or used on some new applications, such as the automatic palmistry for fortune telling.

Part IV: Conclusions

10 CONCLUSIONS AND FURTHER RESEARCH

10.1 Conclusions

In this study, we proposed a framework for high quality palmprint acquisition and effective palm line extraction. Our framework is designed in two layers of abstraction: Layer 1 is the *Palmprint Acquisition Device* composed of a user interface and sensor, while Layer 2 is a *Line Based Recognition Engine* which carries out image preprocessing, feature extraction and matching of palm lines.

Layer 1: Palmprint Acquisition Device

The main objective of Layer 1 is to design a high quality palmprint acquisition device with a comfortable user interface. We performed the design and requirement analysis on the palmprint acquisition device in this layer. Next, the internal components, user interfaces and the appearance design are investigated. Then it evaluates the performance of the palmprint acquisition device.

A flat platen surface is specially designed as the user interface for the guidance of the palm alignment during acquisition. A more reliable coordination system is achieved by this design so that more stable palmprint features can be obtained using this new user interface design. Although there are different types of biometric sensors available, we found that the CCD sensor with video frame grabber is the most appropriate one in the palmprint acquisition. We emphasized that the lighting environment, especially the

uniform illumination on the palm, is very important in our framework. So, the optical system is fabricated in a controlled environment in order to minimize the effects of ambient light. We performed experiments to test different types of light source, they are: 1) bulb light, 2) white Light Emitting Diode (LED), 3) cold cathode fluorescent (CCF) lamp, and 4) ring shape fluorescent light. Experimental result shows that only the ring shape fluorescent light can provide the best image quality. The palmprint acquisition device can be operated in 150 dpi resolutions, which is able to obtain all the line features from a palm including principal lines, wrinkles and ridges. It is a durable device which has acquired more than 9,400 palmprint images. We are proud to announce the first civilian online palmprint identification system using the proposed palmprint acquisition system for the palmprint acquisition, which is placed at our laboratory entrance for access control since early March 2003. We believed that our acquisition device can be applied in different application domains including the employee management system and physical access control in the near future.

Layer 2: Line Based Recognition Engine

The main objective of Layer 2 is to design a line based recognition engine that based on the palm lines' structural information. It consists of palmprint signal preprocessing, line feature extraction, and line matching.

In the palmprint signal preprocessing stage, some typical edge detection techniques are investigated. From the experimental results, we found that traditional edge detection techniques cannot extract the palm lines effectively because they are all dependent on the global thresholding technique which is not good for the peculiarity line features of a palm. Adaptive thresholding technique, on the other hand, uses a local window to calculate the thresholding value which is performed better on palm line extraction. Only few steps are needed to obtain all details of palm lines. It provides a solid foundation on the successive

lines feature extraction. In the line feature extraction stage, we proposed a method to promote the accuracy of palm line extraction by incorporating knowledge. We first defined the bifurcation points and end points in a palmprint image to create sets of line segments and structures, and finally formed the complete palmprint structural map. Next, we defined the properties and types of line segment. Then, we also defined searching strategies to exploit the whole structural information of a palm, they are: 1) single bifurcation, 2) complex bifurcation – looping, 3) complex bifurcation – crossover, 4) complex bifurcation – other cases, and 5) single line. We defined some rules on line operations for joining or splitting a line so as to correct the problems occurred. Different problems are solved, such as: 1) isolated points, 2) broken lines, 3) short bifurcation lines, 4) crossed over lines, and 5) looped lines. We can conclude that by incorporating the above rules with the knowledge from the palm line structures, the major palm lines can be extracted effectively. In line matching stage, the identification template is compared with the master templates such that the system determines whether they are come from the same person. We first define a bounding box to limit the searching area. Next, we divide each line into shorter lines called nodes and perform angle comparisons node by node to get the similarity scores. We introduced the idea of point shifting and node shifting in order to take care of the translation and rotation problems. In addition, we employed a bidirectional matching scheme to further enhance the results. The experimental results show that our palm line extraction framework is effective. We think that the line based approach can make contributions on the classification of palmprint, or be used on some new applications, such as the automatic palmistry for fortune telling.

10.2 Further Research

The further research direction on the palmprint acquisition device should be focused on the liveness detection mechanism during palmprint acquisition. In addition, the size of our device is another concern.

For palm line extraction, we may exploit more knowledge from the palmprint structures so that more rules can be defined to improve the correct line extraction. In addition, we can try to use dynamic selection of parameters such as windows size and the constant value, from the binarization process so that the preliminary line map of a palmprint image can be enhanced. From the experiments, although the recognition rate is not satisfied for a personal identification system, we think that the line based approach can make contributions on the classification of palmprint, or used on some new applications such as the automatic palmistry for fortune telling.

REFERENCES

[1] A. Jain, R. Bolle and S. Pankanti (eds.), *Biometrics: Personal Identification in Networked Society*, Kluwer Academic Publishers, Boston Hardbound, January 1999.

[2] P. Fox, "Want to Save Some Money? Automate Password Resets", retrieved 11 May 2003, from http://www.computerworld.com/securitytopics/security/story/0,10801,61964,00.html

[3] G.S.K. Fung, R.W.H. Lau and J.N.K Liu, "A Signature Based Password Authentication Method", in *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC'97)*, Orlando, Florida, USA, pp. 631-636, October, 1997.

[4] D. Zhang (Ed.), *Biometric Solutions for Authentication in an E-world*, Kluwer Academic Publishers, Boston Hardbound, July 2002.

[5] D.G. Joshi, Y.V. Rao, S. Kar, V. Kumar and R. Kumar, "Computer-Vision-Based Approach to Personal Identification Using Finger Crease Pattern", *Pattern Recognition*, vol. 31, no. 1, pp. 15-22, 1998.

[6] R. Sanchez-Reillo, C. Sanchez-Avilla and A. Gonzalez-Marcos, "Biometric Identification through Hand Geometry Measurements", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no 10, pp.1168-1171, 2000.

[7] R. Wildes, "Iris Recognition: an Emerging Biometric Technology", *Proceeding of the IEEE*, vol. 85, no. 9, pp. 1349-1363, 1997.

[8] A.K. Jain, L. Hong and R. Bolle, "On-line Fingerprint Verification", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no 4, pp. 302-314, 1997.

[9] B. Miller, "Vital Signs of Identity", *IEEE Spectrum*, vol. 31, no. 2, pp. 22-30, February 1994.

[10] D. Zhang, *Automated Biometrics: Technologies and Systems*, Kluwer Academic Publishers, Boston Hardbound, May, 2000.

[11] Identix Press Releases, "Mexican Government Adopts FaceIt® Face Recognition Technology to Eliminate Duplicate Voter Registrations in Upcoming Presidential Elections", retrieved 15 May 2003, from http://www.shareholder.com/identix/ReleaseDetail.cfm?ReleaseID=53264

[12] *Biometrics Market Report 2003-2007*, International Biometric Group, 2003.

[13] S. Pankanti, R.M. Bolle, A. Jain, "Biometrics: The Future of Identification", *IEEE Computer*, vol. 33, no. 2, pp. 46-49, 2000.

[14] D. Zhang, "Biometrics Technologies and Applications" *in Proceedings of First International Conference on Image and Graphics*, Tianjin, China, pp.42-49, August 16-18, 2000.

[15] A.K. Jain and S. Pankanti, "Automated Fingerprint Identification and Imaging Systems", *Advances in Fingerprint Technology*, 2nd Ed. (H. C. Lee and R. E. Gaensslen), Elsevier Science, New York, 2001.

[16] A. K. Jain, A. Ross and S. Prabhakar, "An Introduction to Biometric Recognition", *IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Image- and Video-Based Biometrics*, vol. 14, no. 1, pp. 4-20, January 2004.

[17] J.G. Daugman, "High Confidence Visual Recognition of Persons by a Test of Statistical Independence", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1148-1161, 1993.

[18] Iridian Technologies, Inc., 2002, *Product Overview*, retrieved 1 May 2003, from http://www.iridiantech.com/products.php

[19] J. Daugman, retrieved 9 May 2003, from http://www.cl.cam.ac.uk/users/jgd1000/

[20] D. McGuire, "Virginia Beach Installs Face-Recognition Cameras", *The* Washington Post, 2002, retrieved 14 May 2003, from http://www.washingtonpost.com/ac2/wp-dyn/A19946-2002Jul3

[21] International Biometric Group's Consumer Response to Biometrics, 2002. retrieved 10 May 2003, from http://www.ibgweb.com/reports/public/reports/facial-scan_perceptions.html

[22] N. Ratha, S. Chen, K. Karu and A.K. Jain, "A Real-Time Matching System for Large Fingerprint Databases", *IEEE Transactions on Pattern Analysis and Machine ntelligence*, vol. 18, no 8, pp. 799-813; 1996.

[23] K. Karu and A.K. Jain, "Fingerprint Classification", *Pattern Recognition*, vol. 29, no. 3, pp. 389-404, 1996.

[24] R. Cappelli, A. Lumini, D. Maio and D. Maltoni, "Fingerprint Classification by Directional Image Partitioning", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 402-421, 1999

[25] D. Maio and D. Maltoni, "Direct Gray-Scale Minutiae Detection in Fingerprints", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 1, pp. 27-40, 1997.

[26] J. Berry, "The history and development of fingerprinting", *Advances in Fingerprint Technology*, (H.C. Lee and R.E. Gaensslen, ed.s), CRC Press, Florida, pp. 1-39, 1994.

[27] INSPASS Project, retrieved 28 May 2003, http://www.panynj.gov/aviation/inspass main.htm.

[28] R. Sanchez-Reillo and C. Sanchez-Acila, "Access Control System with Hand Geometry Verification and Smart Cards", *IEEE Aerospace and Electronics Systems Magazine*, vol. 15, no. 2, pp. 45-48, 2000.

[29] Fujitsu Laboratories Limited, "Biometric Mouse with Palm Vein Pattern Recognition Technology", retrieved 4 July, 2003, from http://pr.fujitsu.com/en/news/2002/08/28.html

[30] Fujitsu Laboratories Limited, "Contactless Palm Vein Pattern Biometric Authentication System", retrieved 5 July, 2003, from http://pr.fujitsu.com/en/news/2003/03/31.html

[31] W. Shu and D. Zhang, "Automated Personal Identification by Palmprint", *Optical Engineering*, vol. 37, no. 8, pp. 2659-2362, 1998.

[32] N. G. Altman, Lipton, Ebert H. Charles., and Michael, "Palm Print Identification System", *US Patent No. 3581282*, 1971.

[33] D. F. Maase, T. F. Sartor, and Idnetix Incorporated, "Electro-optic palm scanner system employing a non-planar platen", *US Patent No. 5528355*, 1996.

[34] E. G. Nassimbene, and International Business Machines Corporation, "Palm print identification", *US Patent No. 4032889*, 1977.

[35] T. F. Sartor, and Identix Incorporated, "Apparatus and method for optically imaging features on the surface of a hand", *US Patent No. 6175407*, 2001.

[36] T. Sekiya and Asahi Kogaku Kogyo Kabushiki Kaisha, "Image detecting apparatus for an individual identifying system", *US Patent No. 5526436*, 1996.

[37] Cross Match Technologies, Inc., 2002. "*Cross Match Technologies' ID 1500*", retrieved 14 May 2003, from http://www.crossmatch.net/products_livescan_id 15.html

[38] Identix Incorporated, 2002, "*TouchPrint™ PRO Full Hand Scanner*", retrieved 14 May 2003, from http://www.identix.com/products/pro_livescan_TPpro.html

[39] NEC Solutions (America), Inc., 2002. *Automated Palmprint Identification System*, retrieved 14 May 2003, from http://www.necsolutions-am.com/idsolutions/ download/palmprint/palmprint.html

[40] Printrak 2001. *Omnitrak 8.0 AFIS/Palmprint Identification Technology*, retrieved 14 May 2003, from http://www.printrakinternational.com/omnitrak.htm

[41] SPEX Forensics, 2002. *PrintQuest Automated Palmprint Identification System*, retrieved 14 May 2003, from http://www.printquest-apis.com/

[42] Biometric Partners Inc., *TPS 300 Touchless Palm Sensor*, retrieved 14 May 2003, from http://www.biometricpartners.com/Touchless/TPS_300/tps_300.html

[43] D. Zhang, W.K. Kong, J. You and M. Wong, "On-line Palmprint Identification", *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1041-1050, 2003.

[44] N. Duta, A.K. Jain and K.V. Mardia, "Matching of Palmprint", *Pattern Recognition Letters*, vol. 23, no. 4, pp. 477-485, 2002.

[45] C.C. Han, H.L. Cheng, C.L. Lin and K.C. Fan, "Personal Authentication Using Palm-print Features", *Pattern Recognition*, vol. 36, no. 2, pp. 371-381, 2003.

[46] W. Shu and D. Zhang, "Palmprint Verification: an Implementation of Biometric Technology", *Proceedings of 14th International Conference on Pattern Recognition*, Brisbane Australia, pp. 219-221, 1998.

[47] J. Chen, C. Zhang and G. Rong, "Palmprint Recognition Using Crease", *Proceedings of 2001 International Conference on Image Processing*, vol. 3, pp. 234-237, 2001.

[48] W. Shi, G. Rong, Z. Bain and D. Zhang, "Automatic Palmprint Verification", *International Journal of Image and Graphics*, vol. 1, no. 1, pp. 135-152, 2001.

[49] W. Shu, "Research on Automatic Palmprint Recognition", *Ph.D. Thesis*, Tsinghua University, 1999.

[50] W.K.A. Kong, "Using Texture Analysis On Biometric Technology For Personal Identification", *M.Phil. Thesis*, The Hong Kong Polytechnic University, 2002

[51] D. Zhang and W. Shu, "Two Novel Characteristics in Palmprint Verification: Datum Point Invariance and Line Feature Matching", *Pattern Recognition*, vol. 32, no. 4, pp. 691-702, 1999.

[52] Smart Identity Card System, The Hong Kong Special Administrative Region Government, retrieved 13 May 2003, from http://www.info.gov.hk/gia/general/200202/26/0226207.htm

[53] Fujitsu, 2002, "*Smallest Fingerprint Sensor is Ideal for Mobile Applications*", retrieved 14 May 2003, from http://www.fme.fujitsu.com/news/summer02/07.html

[54] Veridicom, 2002, "*Fingerprint Authentication Systems*", retrieved 14 May 2003, from http://www.veridicom.com/products/overview.htm

[55] Jill Gilbert, "*How to Do Everything with Your Scanner*", McGraw-Hill, Osborne London, 2001

[56] Eastman Kodak Company, 2001. "*Technical Overview: CCD Technology*", retrieved 11 June 2003, from http://www.kodak.com/global/en/service/professional/tib/tib4131.jhtml

[57] Conexant Systems, Inc., 2002, retrieved 14 May 2003, from http://www.conexant.com/products/entry.jsp?id=272

[58] Gerald C. Holst, *CCD Arrays, Cameras, and Displays*, JCD Publishing and SPIE Optical Engineering Press, USA, 1998

[59] M. W. Burke, *Handbook of Machine Vision Engineering*, Chapman & Hall, Great Britian, 1996

[60] R. Wildes, J.C. Asmuth, G.L. Green, S.C. Hsu, R.J. Kolczynski, J.R. Matey and S.E. McBride, "A System for Automated Iris Recognition", *In Proceedings of the IEEE Workshop on Applications of Computer Vision*, pp. 121-128, 1994.

[61] The Imaging Source, 2001, "Induction to Optics and Lenses", *The Imaging Source*, retrieved 14 May 2003, from http://www.theimagingsource.net

[62] Campbell, James B., *Introduction to Remote Sensing*, 3rd Edition, Guilford Press, New York, 2002.

[63] Warren J. Smith, *Modern Optical Engineering : The Design of Optical Systems*, 3rd edition, McGraw-Hill Professional, New York, 2000

[64] Warren J. Smith, *Modern Lens Design : A Resource Manual*, McGraw-Hill, New York, 1992

[65] Keith Jack, *Video Demystified : A Handbook for the Digital Engineer*, 3rd edition, LLH Technology Pub., Eagle Rock, VA, 2001.

[66] Eugene Hecht, *Optics*, Addison-Wesley Publishing, Massachusetts, 2002

[67] Robert E. Fischer, Biljana Tadic-Galeb, *Optical System Design*, McGraw Hill, New York, 2000

[68] C. S. Law, 2000, "Polarization Effects on MTF", retrieved 14 May 2003, from http://www-ise.stanford.edu/class/psych221/00/cslaw/Polarization_Effects_files/frame.htm

[69] W. Li, D. Zhang and Z. Xu, "Palmprint identification by Fourier Transform", *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 16, no. 4, pp. 417-423, 2002.

[70] J. You, W. Li and D. Zhang, "Hierarchical Palmprint Identification via Multiple Feature Extraction", *Pattern Recognition*, vol. 35, no. 4, pp. 847-859, 2002.

[71] K.I. Laws, "Textured Image Segmentation", *Ph.D. Thesis*, University of Southern California, 1980.

[72] J.A. Noble, "Finding Corners", *Image and Vision Computing Journal*, vol. 6, no. 2 pp.121-128, 1988

[73] D.P. Huttenlocher, G.A. Klanderman and W.J. Rucklidge, "Comparing Images Using the Hausdorff Distance", *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 15, no. 9, pp. 850-863, 1993.

[74] T.Y. Young and K.S. Fu, *Handbook Pattern Recognition and Image Processing*, Academic Press, London, 1986.

[75] R.C. Gonzalez, R.E. Woods, *Digital Image Processing*, New York, 1993.

[76] B. Jahne, *Digital Image Processing*, 2nd ed., Springer-Verlag, Berlin, 1993.

[77] J. Canny, "Computational Approach to Edge Detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679-698, Nov. 1986.

[78] J.R. Parker, "Gray Level Thresholding in Badly Illuminated Images", *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 13, no. 8, pp. 813-819, 1991.

[79] Ø.D. Trier and T. Taxt, "Improvement of 'Integrated Function Algorithm' for Binarization of Document Images", *Pattern Recognition Letters*, vol. 16, no. 3, pp. 277-283, March 1995.

[80] A.S. Abutaleb, "Automatic Thresholding of Gray-Level Pictures Using Two-Dimensional Entropy", *Computer Vision, Graphics and Image Processing*, vol. 47, pp. 22-32, 1989.

[81] J.N. Kapur, P.K. Sahoo, and A.K.C. Wong, "A New Method for Gray-Level Picture Thresholding Using the Entropy of the Histogram", *Computer Vision, Graphics and Image Processing*, vol. 29, pp. 273-285, 1985.

[82] J. Kittler and J. Illingworth., "Minimum Error Thresholding", *Pattern Recognition*, vol. 19, no. 1, pp. 41-47, 1986.

[83] C.K. Chow and T. Kaneko, "Automatic detection of the left ventricle from cineangiograms", *Computers and Biomedical Research*, vol. 5, pp. 388-410, 1972.

[84] W. Niblack, *An Introduction to Digital Image Processing*, Prentice Hall International, Englewood Cliffs, New Jersey, 1986.

[85] J. Bernsen, "Dynamic Thresholding of Grey-level Images", *Proceedings 8th International Conference on Pattern Recognition*, pp. 1251-1255, Paris, 1986.

[86] Y. Nakagawa and A. Rosenfeld, "Some Experiments on Variable Thresholding", *Pattern Recognition*, vol. 11, no. 3, pp. 191-204, 1979.

[87] L. Eikvil, T. Taxt, and K. Moen, "A Fast Adaptive Method for Binarization of Document Images", *Proceedings the 1st International Conference on Document Analysis and Recognition*, Saint-Malo, France, pp. 453-443, 1991.

[88] K.V. Mardia and T.J. Hainsworth, "A Spatial Thresholding Method for Image Segmentation", *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 10, no. 6, pp. 919-927, 1988.

[89] T. Taxt, P.J. Flynn, and A.K. Jain, "Segmentation of Document Images", *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 11, no. 12, pp. 1322-1329, 1989.

[90] S.D. Yanowitz and A.M. Bruckstein, "A New Method for Image Segmentation", *Computer Vision, Graphics and Image Processing*, vol. 46, no. 1, pp. 82-95, April 1989.

[91] J.M. White and G.D. Rohrer, "Image Thresholding for Optical Character Recognition and Other Applications Requiring Character Image Extraction", *IBM J. Research and Development*, vol. 27, no. 4, pp. 400-411, July 1983.

[92] N. Otsu, "A Threshold Selection Method from Gray Level Histograms", *IEEE Transactions On Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62-66, January 1979.

[93] X. Wu, K. Wang and D. Zhang, "Fuzzy Directional Element Energy Feature (FDEEF) Based Palmprint Identification", *International Conference on Pattern Recognition*, vol. 1, pp. 10095-10098, 2002.

[94] A. Datta and S.K. Parui, "A Robust Parallel Thinning Algorithm for Binary Images", *Pattern Recognition*, vol. 27, no. 9, pp. 1181-1192, 1994.

[95] L. Lam, S.W. Lee, C.Y. Suen, "Thinning Methodology – A Comprehensive Survey", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 9, pp. 869-885, 1992.

[96] C.J. Hilditch, "Linear Skeletons from Square Cupboards", *Machine Intelligence*, vol. 4, pp. 403-420, 1969.

[97] M. Kass and A. Witkin, "Analyzing Oriented Patterns", *Computer Vision, Graphic Image Processing*, vol. 37, no. 4, pp. 362-385, 1987.

[98] M. Kawagoe and A. Tojo, "Fingerprint Pattern Classification", *Patter Recognition*, vol. 17, no. 3, pp. 295-303, 1984.

[99] A. R. Rao, *A Taxonomy for Texture Description and Identification*, Springer-Verlag, New York, 1990

[100] Fingerprint Verification Competition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no 3, pp. 402-412, 2002.

BRIEF CURRICULUM VITAE

Michael Wong is an Mphil student at the Department of Computing, The Hong Kong Polytechnic University (PolyU). He received his Bachelor of Arts in Computing in 2001, from PolyU with first class honors and awarded The Reuter Foundation Scholarship in 2001. In his Mphil study, he received two years of Tuition Scholarship for Research Postgraduate Studies awarded by PolyU in 2001/2002 and 2002/2003. His research interests are primarily in Biometrics and Pattern Recognition.

Publications

- Michael Wong and David Zhang, "Knowledge Based Palm Lines Extraction with Adaptive Thresholding", *Proceedings of the 3rd IASTED International Conference on Visualization, Imaging, and Image Processing*, 8-10 September 2003, pp. 692-697, Benalmádena, Spain.
- David Zhang, Wai-Kin Kong, Jane You and Michael Wong, "On-line palmprint identification", *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1041-1050, 2003.
- Michael Wong and David Zhang, "Palmprint Acquisition Device: Requirement Analysis and Implementation", *Proceedings of the ISCA 12th International Conference on Intelligent and Adaptive Systems and Software Engineering*, 9-11 July, 2003, pp. 75-78, San Francisco, CA. USA.

- Michael Wong and David Zhang, "Design and Implementation of a High Quality Palmprint Acquisition Device", *Proceedings of ACM Postgraduate Research Day*, pp. 39-44, January 2003, Hong Kong.

Working Paper

David Zhang, Wai-Kin Kong, and Guangming Lu, and Michael Wong, "Palmprint Identification System for Civil Applications", will be submitted to the *IEEE Transactions on Systems, Man, and Cybernetics*.

Patent

Dapang Zhang, Guangming Lu, Wai-Kin Adams Kong and Michael Wong, "Apparatus for Capturing a Palmprint Image", Pending US Patent Application No. US PTO 10/253,912 filed, August 2002

Project Awards

- Silver Medal, Palmprint Identification System, *Seoul International Invention Fair 2002*, 4-8th December 2002, Pacific Hall, COEX World Trade Center, Seoul, Korea.
- Consumer Product Design Certificate of Merit, Palmprint Identification System, *2003 Hong Kong Awards for Industry*, 27th October 2003, Hong Kong.
- Gold Medal, Palmprint Identification System, *14th National Inventions Exhibition of China*, 23-28th October 2003, Xiamen, China.
- Special Gold Award, Palmprint Identification System, *14th National Inventions Exhibition of China*, 23-28th October 2003, Xiamen, China.