PALMPRINT RECOGNITION BASED ON 2-D GABOR FILTERS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

BARIŞ KONUK

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

JANUARY 2007

Approval of the Graduate School of (Name of the Graduate School)

_____

Prof. Dr. Canan ÖZGEN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

_____

Prof. Dr. İsmet ERKMEN
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

_____

Assoc Prof. Dr. Gözde BOZDAĞI AKAR
Supervisor

Examining Committee Members (first name belongs to the chairperson of the jury and the second name belongs to supervisor)

Prof. Dr. Murat AŞKAR (METU, EE) _____

Assoc Prof. Dr. Gözde BOZDAĞI AKAR (METU, EE) _____

Prof. Dr. Volkan ATALAY (METU, CENG) _____

Assoc. Prof. Dr. Aydın ALATAN (METU, EE) _____

Assist. Prof. Dr. Çağatay CANDAN (METU, EE) _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : Barış KONUK

Signature :

ABSTRACT

PALMPRINT RECOGNITION BASED ON 2-D GABOR FILTERS

KONUK, Barış
M.S., Department of Electrical and Electronics Engineering

Supervisor : Assoc. Prof. Dr. Gözde BOZDAĞI AKAR

January 2007, 76 pages

In this thesis work, a detailed analysis of biometric technologies has been done and a new palmprint recognition algorithm has been implemented. The proposed algorithm is based on 2-D Gabor filters. The developed algorithm is first tested on The Hong Kong Polytechnic University Palmprint Database in terms of accuracy, speed and template size. Then a scanner is integrated into the developed algorithm in order to acquire palm images; in this way an online palmprint recognition system has been developed. Then a small palmprint database is formed via this system in Middle East Technical University. Results on this new database have also shown the success of the developed algorithm.

Keywords: Biometrics, Palmprint Recognition, Two-dimensional Gabor Filter

ÖZ

2-B GABOR SÜZGEÇ TABANLI AVUÇ İZİ TANIMA

KONUK, Barış
Yüksek Lisans Tezi, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi : Doç. Dr. Gözde BOZDAĞI AKAR

Ocak 2007, 76 sayfa

Bu tez çalışmasında, biyometrik teknolojilerin detaylı analizi yapılmış ve yeni bir avuç izi tanıma algoritması gerçeklenmiştir. Önerilen algoritma 2-B Gabor süzgeç tabanlıdır. Geliştirilen algoritma ilk olarak Hong Kong Polytechnic Üniversitesi avuç izi veritabanında doğruluk, hız ve şablon büyüklüğü açılarından test edilmiştir. Daha sonra avuç ayası resimleri tedarik etmek amacıyla, bir tarayıcı geliştirilen algoritmaya entegre edilmiş, böylece bilgisayar kontrollü avuç izi tanıma sistemi geliştirilmiştir. Daha sonra, bu sistem ile Orta Doğu Teknik Üniversitesi'nde küçük bir avuç izi veritabanı oluşturulmuştur. Bu yeni veritabanı üzerindeki sonuçlar da geliştirilen algoritmanın başarısını göstermiştir.

Anahtar Kelimeler: Biyometrik, Avuç İzi Tanıma, 2 Boyutlu Gabor Süzgeç

To My Parents and To My Brother

ACKNOWLEDGMENTS

I would like to thank Assoc Prof. Dr. Gözde Bozdağı Akar for her valuable supervision, support and tolerance throughout the development and improvement of this thesis.

I am grateful to Metin Şengül and Volkan İpek for their support throughout the development and the improvement of this thesis. I am also grateful to Aselsan Electronics Industries Inc. for the resources and facilities that I use throughout thesis.

Thanks a lot to all my friends for their great encouragement and their valuable help to accomplish this work.

Finally, I would like to thank my parents for bringing up and trusting in me, and to my brother, whom I love in deep, for his valuable friendship.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

FMR : False Match Rate

FAR : False Acceptance Rate

FNMR : False Non-Match Rate

FRR : False Rejection Rate

FTER : Failure to Enroll Rate

ROC : Receiver Operating Characteristics

EER : Equal Error Rate

DPI : Dots per Inch

METU : Middle East Technical University

CHAPTER I

INTRODUCTION

In our daily lives, there is a frequent need in identifying people correctly and verifying their identities. To illustrate, reliable identification mechanisms are required when people board an aircraft, perform financial operations, desire to enter secure places etc. For higher efficiency and increased security, this identification mechanism should be automated. Obviously, high accuracy is required during the identification and this hardens the automation of identification. But once automated, it gives us the opportunity that tasks performed by computers and other devices can be widened and this results in easing our lives. It is here worth noting that, tasks performed by these devices are based on two separate mechanisms, namely authentication and authorization. Authentication is known as identity verification, whereas authorization defines particular rights of authenticated people. Therefore, authorization follows authentication.

There are three fundamental modes of authentication:

1. What you know, which generally refers to information to be kept secret such as passwords and pass phrases or some non-secret private information such as mother's maiden name.
2. What you have, which generally refers to physical possessions such as keys and smartcards.
3. What you are, which generally refers to biometrics, physical appearances or behavioral characteristics of individuals such as fingerprint, hand geometry and signature.

These entire authentication methods depend on the same basic principle, which can be summarized as follows [1]:

- User provides an authenticator, a data item that can not be provided by anyone else.
- Authentication system contains a verifier, a data item that can verify the correctness of the authenticator.
- Authentication system uses a verification procedure, an algorithm that compares an authenticator with a verifier.
- There is generally a base secret, a data item in user's possession that produces the authenticator.

Among these authentication modes, knowledge based authentication, mode 1, is the most common and the easiest authentication mode. Because of being the easiest method, the first computer-based authentication system, which was implemented in the early 1960s as a part of the Compatible Time Sharing System (CTSS) at the Massachusetts Institute of Technology (MIT), was based on this method. [1] In this mode, individuals are asked to provide passwords which are used as the authenticator, the verifier and the base secret. Verification procedure for this method is simply the string comparison of the authenticator and the verifier. In this mode, the reliability of the authentication depends on the strength of passwords; therefore, passwords provided by users are expected to be secret and hard-to-guess. Nevertheless, users usually tend to select weak passwords, which are easy-to-guess, in order to remember easily when required. As a precaution to these weak passwords, users are generally forced to select stronger passwords. However, since stronger passwords are hard to remember, users respond by posting their hard-to-guess passwords on their computer screens. Additionally, users may share their passwords. When this is the case, users will be authenticated as different users. In brief, this mode of authentication is relatively easier to implement, nonetheless, it can be said that this mode is not proper for high security applications.

The second authentication method, possession based authentication, is considered to be a more reliable method than knowledge based authentication. In this mode, users have smartcards, keys or tokens as the name implies. These devices generally use a base secret such as the current time or an internal counter in order to produce the

authenticator. Depending on the technology used, special verification procedures implemented in the authentication system are applied to the authenticator and the verifier is obtained. Then the system can perform authentication by comparing the verifier and the authenticator. Nonetheless, users may share their smartcards, which result in wrong authentications, as it is the case in the knowledge based authentication. Even worse, tokens can be lost or stolen. But a major benefit of this authentication method is that the owner of the card or token will be aware of the fact that the card is lost, which is not the case in knowledge based authentication. Nevertheless, the authentication system has no chance of realizing that the authenticator is produced by a stolen device unless the owner of the stolen device warns the authentication system administrators.

The last authentication mode, biometrics, is known to be the most reliable method. Biometrics is defined as the science of identifying, or verifying the identity of, a person based on physiological or behavioral characteristics. [2] The term, a biometric, is also used to refer a specific mode of recognizing people (thus fingerprint and face recognition are two biometrics) or examples of the specified characteristics being recognized (a fingerprint and a face image are both biometrics) [2]. The biometric reading collected from an individual via a biometric sensor serves as an authenticator for this mode. Biometric readings collected previously from the same individual serve as verifier. Some algorithm implemented in the authentication system, measures the distance between the authenticator and the verifier and decides whether the authenticator matches the verifier. This algorithm serves as the verification procedure. Since passwords are not employed and biometrics can not be shared or stolen, biometrics authentication mode is not subject to threats mentioned above. Unless physical characteristics or personal traits of individuals change severely, biometric authentication can be used reliably.

There are many biometric authentication methods. Among these methods, finger-scan, facial-scan, iris-scan and voice-scan are the most widely used biometric authentication methods. Nevertheless, there is no single best biometric technology, that is, each of these biometric technologies has both advantages and disadvantages, which will be detailed later.

1.1 Description and Outline of the Thesis

In this work, a palmprint recognition algorithm based on 2-D Gabor filters is implemented in MATLAB® environment. Nevertheless, built-in functions in MATLAB® Image Processing Toolbox® are almost not utilized in order to develop a platform-independent algorithm. The developed algorithm is first tested on The Hong Kong Polytechnic University Palmprint Database. Results obtained on this database have been investigated and some possible improvements in the palmprint acquisition block have been noted. Considering these possible improvements, a complete palmprint recognition system has been implemented by integrating a palmprint acquisition block employing a scanner into the developed algorithm in MATLAB® environment. A small palmprint database is formed in Middle East Technical University (METU) via this system and the algorithm is also tested on this database. The implementation details of the design and the results obtained on these databases are explained in detail.

This thesis is organized as follows: In Chapter II, biometrics, the emerging and reliable authentication method, is discussed in detail. In this chapter, key metrics used in the evaluation of biometric systems have been defined, and advantages and disadvantages of some leading biometric technologies currently being used are mentioned. Moreover, advantages of palmprint as a biometric have been explained.

In Chapter III, brief information is given about The Hong Kong Polytechnic University Palmprint Database, the most commonly used palmprint database which is also used in this thesis. Furthermore, some of the palmprint recognition methods in the literature are described and results obtained in these studies are presented.

In Chapter IV details of the developed palmprint recognition algorithm are given. In this chapter, the algorithm is divided into three sub-blocks and each sub-block is detailed along with the discussions related to the effects of different parameters.

In Chapter V, results of the developed algorithm on The Hong Kong Polytechnic University Palmprint Database are presented. Obtained results are investigated from different side of views and factors affecting results are discussed. Moreover, in order to verify that the developed algorithm is a generic palmprint recognition algorithm, results obtained on the database, which is formed in METU by the integration of a scanner, which is used as a palmprint acquisition device, and the developed algorithm are mentioned.

Finally, the developed algorithm and results obtained on both databases are summarized. Moreover, some strong and weak points of the algorithm together with some possible improvements on palmprint recognition system are discussed in Chapter VI.

CHAPTER II

BIOMETRICS IN AUTHENTICATION

Because biometric based authentication is emerging as a powerful method for reliable authentication, which is of great importance in our lives, biometrics is becoming increasingly popular. In 2001, the highly respected MIT Technology Review announced biometrics as one of the "top ten emerging technologies that will change the world" [1]. Also Rick Norton, the executive director of the International Biometric Industry Association (IBIA), pointed out the increase in biometric revenues by an order of magnitude over the recent years. Biometric revenues, which were $20 million in 1996, increased by 10 times and reached $200 million in 2001. Rick Norton expects a similar increase in biometric revenues in next 5 years period, from 2001 to 2006, thereby expecting them to reach $2 billion by 2006[1]. Similarly, International Biometric Group, a biometric consulting and integration company in New York City, estimate biometric revenues to be around $1.9 billion in 2005[1].

2.1 Properties of Biometrics

Researchers noticing the increase in biometric revenues are trying to develop better algorithms for existing biometrics and\or to find new biometrics for authentication. Whether new or existing, all practical biometrics should possess five properties described below [2]:

1. Universality: All individuals should possess the biometric characteristics.
2. Uniqueness: The biometric characteristics of different individuals should not be the same.
3. Permanence: The biometric characteristics of individuals should not change severely with the time.

4. Collectability: The biometric characteristics should be measurable with some practical device.
5. Acceptability: Individuals should not have objections to the measuring or collection of the biometric.

2.2 Biometric System Block Diagram

After the biometric that is to be utilized is decided, the question how a biometric system can be implemented naturally arises. Figure 2.1 shows the general block diagram of a biometric system. As shown in Figure 2.1, biometric systems generally consist of the following components:

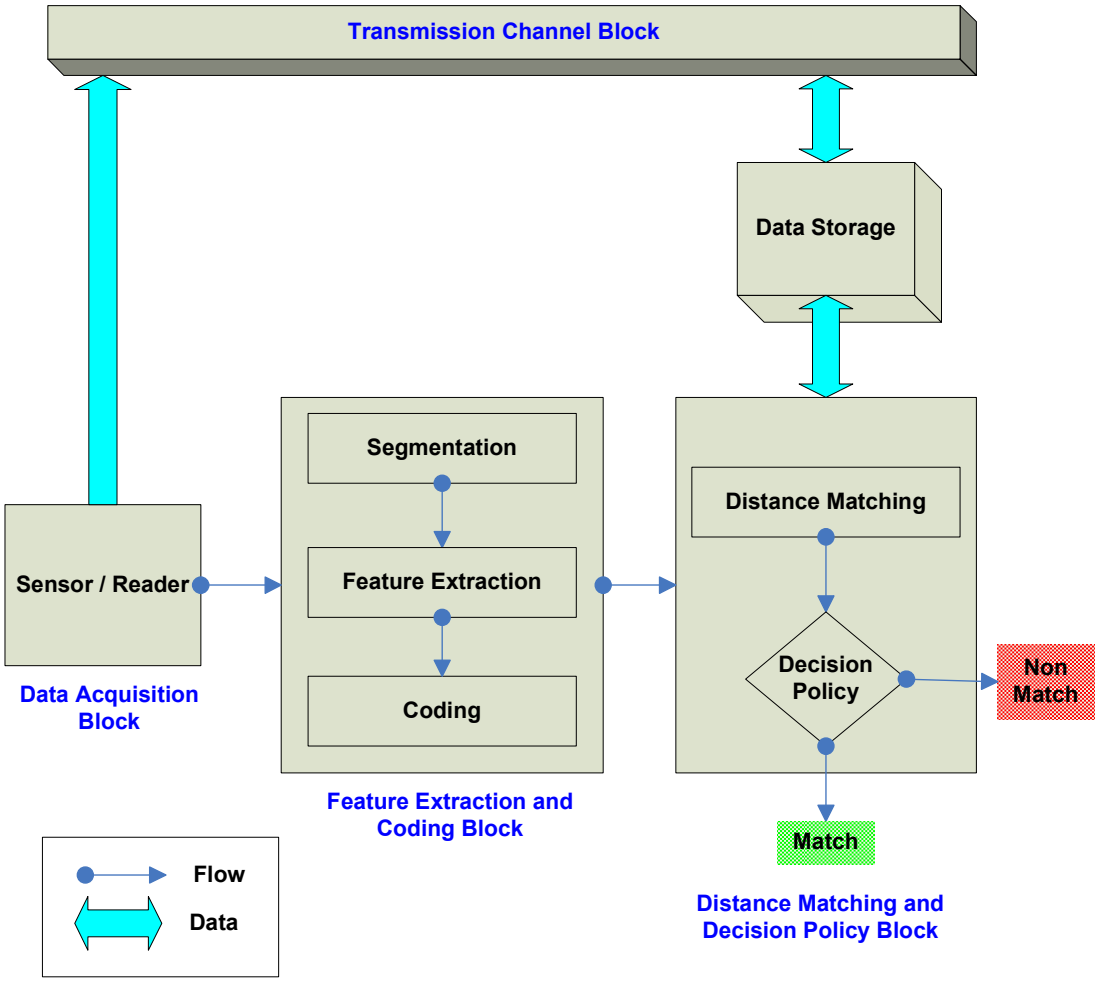

Figure 2.1 General Block Diagram of a Biometric System

- Data Acquisition Block: This is the block in which biometric data is captured and is transferred to feature extraction and coding block. The biometric data may also be compressed in this block, especially when the data acquisition is performed at a remote location.
- Transmission Channel Block: This is an optional block in the sense that some biometric systems do not consist of this block. Although transmission channels are internal to the device in self-contained systems, some biometric systems may be distributed and may have central data storage and many remote data acquisition points. The transmission channel for distributed systems might be a local area network (LAN), a private Intranet, or even the Internet. [1]
- Feature Extraction and Coding Block: This is the block in which acquired biometric sample is processed. Processing consists of segmentation, the process of separating relevant biometric data from background information, and feature extraction, the process of locating and extracting desired biometric data. After segmentation and feature extraction, a biometric template, a mathematical representation of the original biometric, is obtained by encoding extracted features.
- Distance Matching and Decision Policy Block: This is the final block in a biometric system, where the final decision is made. The biometric template obtained in feature extraction and coding block is compared to one or more templates in the data storage by selected matching algorithm, which determines the degree of similarity between compared templates. The final decision is usually made based on the result of the matching algorithm and empirically determined thresholds.

2.3 Verification and Identification

The most important distinction in biometrics is between verification and identification. Verification systems verify or reject users' identity. In verification

systems, the user is requested to prove that he/she is the person he/she claims to be. Therefore; the user should first claim an identity by providing a username or an ID number. After claiming the identity, the user provides a biometric data to be compared against his or her enrolled biometric data. The biometric system then returns one of two possible answers, verified or not verified. Verification is usually referred to as 1:1 (one-to-one), since the biometric data provided by the user is only compared against the enrolled biometric data of the person that the user claims to be. Identification systems, on the other hand, try to identify the person providing the biometric data. In identification systems, the user is not required to claim an identity; which is not the case in verification systems, instead he/she is only requested to provide a biometric data. Another difference of identification from verification is that user's biometric data is compared against a number of users' biometric data. Therefore; identification is generally referred as 1:N (one-to-N or one-to-many). Then the system returns an identity such as a username or an ID number.

2.4 Evaluation of Biometric Systems

Once a biometric system is designed, its performance and accuracy should be evaluated in order to decide whether the designed system is sufficient. To do this, there are three key measurements in biometrics, which are false match rate (FMR), false non-match rate (FNMR) and failure-to-enroll rate (FTER). None of these performance metrics is sufficient to assess the performance of used technology. In fact, all three key measurements, which are detailed below, are necessary to see how well the biometric system performs.

1. False Match Rate (FMR): FMR is defined as the probability that a user's template will be incorrectly matched to a different user's template. FMR describes the likelihood of an imposter beating a biometric system by being matched as someone other than him or herself. False matches may be due to similarities of biometric characteristics of two different users, so that the system finds a high degree of correlation between the user's templates. Here, it may be thought that FMR can be decreased by adjusting thresholds used in

the decision policy, which determine the necessary correlation level for two templates to be matched. Nevertheless, FMR and FNMR are inversely proportional; therefore when FMR is decreased by adjusting thresholds, FNMR will be increased. It is here worth noting that, FMR is the most important key measurement in systems requiring high level of security, such as entry to a weapons facility or a bank vault. FMR is sometimes referred as the false acceptance rate (FAR), because the user is generally accepted into a building, an application, or a resource after a successful match. [3]

2. False Non-match Rate (FNMR): FNMR is defined as the probability that a user's template will be incorrectly judged to not match his or her enrollment template. False non-matches may occur because there is not sufficiently strong correlation between a user's biometric data and his template in data storage. False non-matches are generally due to changes in a user's biometric data and/or changes in environmental conditions. The way user presents biometric data to the system may also be a reason of non-matches. FNMR can be decreased by adjusting threshold; nonetheless this will result in an increase in FMR as mentioned above. Although FMR seems more important than FNMR in many applications, particularly in high security applications; a system with a FMR of 0%, but a FNMR of 50%, is highly secure but unusable. FNMR is sometimes referred as false rejection rate (FRR), because an authorized user is incorrectly denied access to an application or a resource in case of a false non-match. [3]

3. Failure-to-Enroll Rate (FTER): Before defining FTER, it is worth giving the definition of enrollment. Enrollment is defined as the procedure in which biometric information supplied by individuals are stored in the data storage for future matching. Depending on the selected biometric characteristic and the required security level, one or more biometric samples may be provided by individuals during enrollment. The third key measurement, FTER, is defined as the probability that an individual will be unable to enroll in a biometric system. There are many causes, related to both the environment and individuals, resulting in failure-to-enroll. Poor lighting in face recognition

applications and background noise in voice recognition applications are examples for the environmental causes of FTER. There are also causes of FTER which are related to individuals. To illustrate, people working in construction tend to wear down their fingerprints. Beside the occupation, age and ethnicity of individuals can also be considered as causes of FTER related to individuals. FTER can be drastically reduced by informing enrollees in order to get best biometric templates. For example, informing individuals about the proper placement of their fingers in a fingerprint recognition application or asking them to lengthen a pass phrase in a voice recognition application will be useful in reducing FTER. In general, ensuring proper placement and interaction with biometric acquisition devices significantly decrease FTER without requiring any changes in the core biometric processes. [3]

Among these three metrics, biometric system designers generally focus on FMR and FNMR in real world. They usually pay less attention to FTER. Therefore, the designers generally assess the performance of used technology via FMR and FNMR. However, both FMR and FNMR are not constant values; instead they are functions of the threshold, T, used in the decision policy block. Let FMR (T) and FNMR (T) represent the functions giving error rates for the threshold T. The two-dimensional curve on which FMR (T) and FNMR (T) are plotted against each other for different values of T is called the Receiver Operating Characteristics (ROC) curve. Therefore; coordinates of any point on the ROC curve can be expressed ROC (T_0) = (FMR (T_0), FNMR (T_0)) for some threshold level T_0. Figure 2.2 demonstrates sample ROC curves.

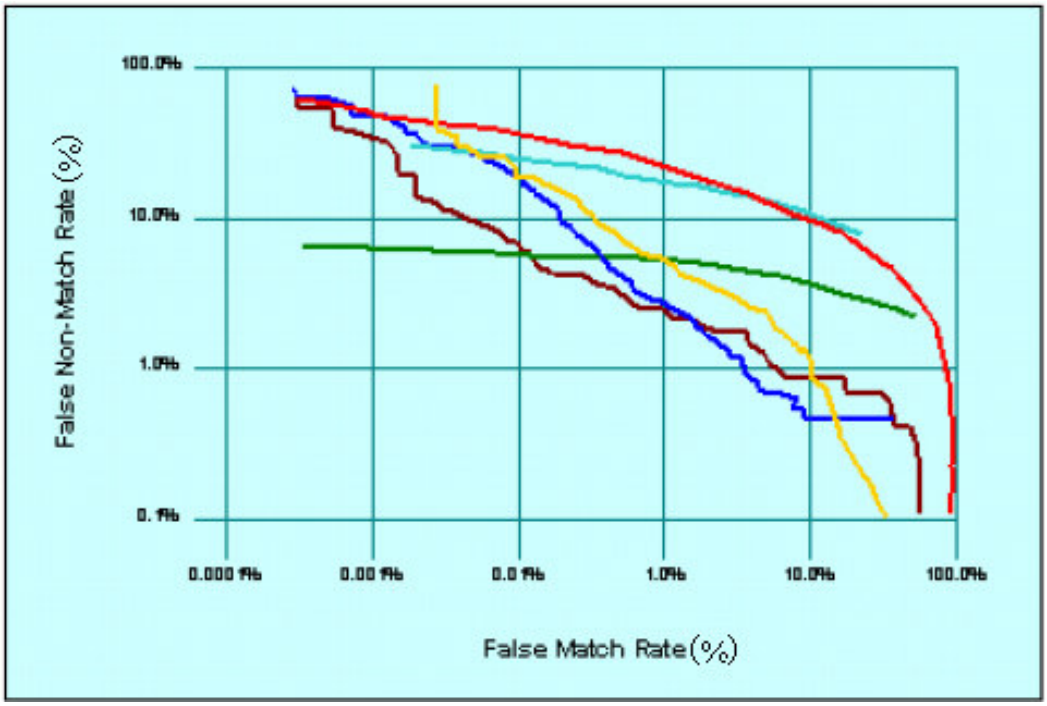

Figure 2.2 Sample ROC Curves [4]

FMR and FNMR, as a function of threshold T, are mapped as:

$$\mathrm{ROC}(T) = (\mathrm{FMR}(T), \mathrm{FNMR}(T)) \rightarrow \begin{cases} (1,0) \text{ as } T \rightarrow 0 \\ (0,1) \text{ as } T \rightarrow \infty \end{cases} \quad \textbf{(2.1)}$$

Equation (2.1) can be explained as follows. When the threshold is low, that is when the degree of similarity required between two templates to be matched is low; FMR is high, because many templates belonging to different individuals may now be considered to be similar enough to be matched. On the other hand, FNMR is low, because templates that belong to same individual most probably have the required degree of similarity; therefore they are not rejected. On the contrary, if the threshold is selected to be high, that is the required degree of similarity is high, templates originating from different individuals are not matched because of not being similar enough. As a result, a low FMR is obtained. However, this may result in high FNMR because templates originating from same individual may now be judged to not match each other.

The point on the ROC curve at which FMR and FNMR are equal is defined as Equal Error Rate (EER). In other words, EER can be explained as the point where the probability of incorrect matching is equal to that of incorrect non-matching. EER is usually considered to represent the overall accuracy of a biometric system and it is generally used in order to compare two different biometric systems. However, EER may sometimes be misleading while selecting a biometric system to be deployed because most of the information about the accuracy of a biometric system is lost by reducing the ROC curve to a single point, EER. EER reflects the accuracy of a system only at a single point. If the system is expected to operate at that single point, where FMR and FNMR are equal, then the system with smaller EER should be chosen to be deployed. Nevertheless, the ability to reject imposters is considered to be more important than the ability to match authorized users in many applications. That is why; many biometric systems are not operated at that single point. Instead they are usually operated at a point where FMR is smaller than FNMR. Therefore, different biometric systems should be compared at the desired operating point in order to decide correctly while selecting the one to be deployed. To illustrate, if it is desired to have a biometric system operating at a FMR of 0.1 %, then the biometric system with the smallest FNMR when FMR is 0.1 % should be preferred. In brief, EER should be considered to give an idea about the system accuracy but it should not be only guideline while deciding the biometric system to be deployed.

Up to here, it is mentioned about the general block diagram of a biometric system and the key metrics being used to evaluate biometric systems. It is here worth mentioning shortly about some leading biometric technologies, including finger-scan, iris-scan, and facial-scan, and about their advantages and disadvantages.

2.5 Leading Biometric Technologies

2.5.1 Finger-Scan

Finger-scan is a well-known biometric technology which is used to identify and verify individuals based on the discriminative features on their fingerprints. Many finger-scan technologies are based on minutiae points, which are irregularities and discontinuities characterizing fingerprint ridges and valleys. [3]

2.5.1.1 Advantages of Finger-Scan Technology

- It is proven to have very high accuracy.
- It does not require complex user – system interaction; therefore little user training is enough to ensure correct placement of fingers.
- It provides the opportunity to enroll up to 10 fingers.

2.5.1.2 Disadvantages of Finger-Scan Technology

- High resolution images are required to be acquired due to the small area of a fingerprint and this results is in more expensive acquisition devices.
- Small percentage of users; elderly populations, manual laborers and some Asian populations; are shown to be unable to enroll in some finger-scan systems according to International Biometric Group's Comparative Biometric Testing. [3]
- As mentioned before, some people may tend to wear down their fingerprints in time because of their physical work.
- Individuals may have objections to collection of their fingerprints because they may have doubts about usage of their fingerprints for forensic applications.

2.5.2 Facial-Scan

Facial-scan is a biometric technology which is used to identify and verify individuals based on the discriminative features on their faces. Nonetheless, it is generally used for identification and surveillance instead of verification. Facial-scan technologies use some of many discriminative features on face such as eyes, nose, lips etc. [3]

2.5.2.1 Advantages of Facial-Scan Technology

- It is the only biometric which provides the opportunity to identify individuals at a distance avoiding user discomfort about touching a device.
- It can use images captured from various devices from standard video cameras to CCTV cameras.

2.5.2.2 Disadvantages of Facial-Scan Technology

- Changes in lighting conditions, angle of acquisition and background composition may reduce the system accuracy.
- The face is a reasonably changeable physiological characteristic. Addition or removal of eyeglasses, changes in beard, moustache, make-up and hairstyle may also reduce the system accuracy.
- In order to take changes in environmental conditions and user appearance into account, facial-scan technologies usually store many templates for each individual and this results in higher memory requirement for each individual compared to many other biometrics.
- Because face of users may be acquired without their awareness, users may have objections to facial-scan deployments.

2.5.3 Iris-Scan

Iris-scan is a biometric technology which is used to identify and verify individuals based on the distinctive features on their irises. Iris-scan technologies use the patterns that constitute the visual component of the iris to discriminate between individuals. [3]

2.5.3.1 Advantages of Iris-Scan Technology

- It is proven to have smallest FMR among all biometrics, therefore; iris is the most suitable biometric for applications requiring highest level of security.
- Iris does not change in time, therefore; it does not require reenrollment which other technologies require after a period of time due to changes in the biometric.

2.5.3.2 Disadvantages of Iris-Scan Technology

- It requires complex user – system interaction, particularly precise positioning of head and eye. Some systems even require that users do not move their head during acquisition.
- Very high resolution images are required to be acquired due to the small area of an iris, therefore; acquisition devices are quite expensive.
- There is a public objection to using an eye-based biometric even though many people are not aware of the fact that infrared illumination is used in iris-scan technology. Were they aware, they might be a much stronger reaction to this technology.

2.5.4 Voice-Scan

Voice-scan is a biometric technology which is used to identify and verify individuals based on the distinctive aspects of their voice. Voice-scan technologies use different

vocal qualities such as fundamental frequency, short-time spectrum of speech and spectograms (time – frequency – energy patterns).[3]

2.5.4.1 Advantages of Voice-Scan Technology

- Various acquisition devices including microphones, land and mobile phones can be utilized and these devices are relatively cheaper than acquisition devices used in other biometrics.
- Users are prompted to select a pass phrase during enrollment and they are asked to repeat the same pass phrase during verification and identification. The probability that imposters guess the correct pass phrase adds an inherent resistance against false matching.

2.5.4.2 Disadvantages of Voice-Scan Technology

- Poor reception quality, ambient noise and echoes may degrade the system accuracy.
- The voice is also a changeable biometric characteristic. Changes in voice due to illness, lack of sleep and mood may reduce the system accuracy.
- Voice-scan is subject to possibility of recording and replay attacks.
- Users are requested to repeat the pass phrase a number of times during enrollment. Therefore, enrollment process in voice-scan is somewhat longer than that in other biometrics.
- Templates in voice-scan usually occupy a number of times more space than those in other biometrics.

2.6 Desired Features in a Biometric and Palmprint

As it is seen, all biometric technologies mentioned above have both advantages and disadvantages. In other words, there is no perfect biometric technology that has no disadvantage. However, it is possible to figure out the desired features in a biometric

technology by inspecting advantages and disadvantages of the biometric technologies above. The list of desired features in a biometric technology is given below:

- High Accuracy
- Zero or very small FTER
- Permanence of biometric in time
- Utilization of cheap acquisition devices
- Resistance to changes in environmental conditions
- No or very little public objection (Acceptability)
- Small template size
- Simple user – system interaction

Inspecting the list above, voice-scan mainly suffers from lower accuracy and higher template size. Facial-scan may not provide the required accuracy due to changes in environmental conditions and user appearance. Although iris is the most reliable biometric, high cost of acquisition devices used in order to scan iris is the biggest handicap of this technology. Finger-scan has a very high accuracy with simple user-system interaction and small template size. Nevertheless, physical work and age may cause people not to have clear fingerprints. Additionally, possible dirt and grease on fingerprints may reduce the system accuracy. Were the area of fingerprint larger, finger-scan technology might suffer less from effects of dirt, physical work and age on fingerprints. Palm, on the other hand, provides a large area for feature extraction and seems to suffer less from factors that reduce the accuracy in finger-scan technology. Moreover, large area of palm enables utilization of low resolution images resulting in cheaper acquisition devices. Furthermore, a very small FTER is expected in palmprint-scan applications because it is easy to correctly place palm on a desired platform. Due to the same reason, it is possible to have a system with simple user – system interaction. Additionally, palmprint-scan is a promising biometric technology to have high accuracy because palmprint is covered with a similar skin as fingerprint. Finally, palmprint-scan technology has high user acceptance which is quite necessary for the technology to spread out.

As it is seen, palmprint possesses the most of desired features therefore; it may be used as a biometric. The next chapter will describe some palmprint recognition algorithms in the literature and will explain results obtained in these algorithms.

CHAPTER III

EXISTING PALMPRINT RECOGNITION ALGORITHMS

Researchers noticing the increase in biometric revenues in last years and realizing the advantages of palmprint scan-technology mentioned in the previous chapter started to develop algorithms to be used in palmprint recognition. Researchers' interest in palmprint recognition algorithms has significantly increased especially in last three years. Due to the fact that the palmprint recognition is a relatively new field of biometrics, there is a problem related to the utilization of a common palmprint database in order to be able to compare the performance of different algorithms. Nevertheless, The Hong Kong Polytechnic University Palmprint Database is the most commonly used palmprint database. It is here worth giving brief information about this database before explaining some of the studies on palmprint recognition.

The Hong Kong Polytechnic University Palmprint Database contains 600 grayscale images corresponding to 100 different palms in Bitmap image format. Palm images have a resolution of 284x384 pixels with 256 gray levels. Six samples from each of these palms were collected in two sessions, where 3 samples were captured in the first session and the other 3 in the second session. The average interval between the first and the second collection was two months. The palmprint images in the database are labeled as "PolyU_xx_N.bmp", where the "xx" is the unique palm identifier (ranges from 00 to 99), and "N" is the index of each palm (ranges from 1 to 6), the palmprints indexed from 1 to 3 are collected in the first session and 4 to 6 in the second session. [5] Figure 3.1 shows a schematic diagram of the online palmprint capture device used to acquire these palm images. The palmprint capture device includes ring source, CCD camera, lens, frame grabber, and A/D (analogue-to-digital) converter. To obtain a stable palmprint image, a case and a cover are used to form a semi-closed environment, and the ring source provides uniform lighting conditions during palmprint image capturing. Also, six pegs on the platform, which

is demonstrated in Figure 3.2, serve as control points for the placement of the user's hands. The A/D converter directly transmits the images captured by the CCD camera to a computer. [6]

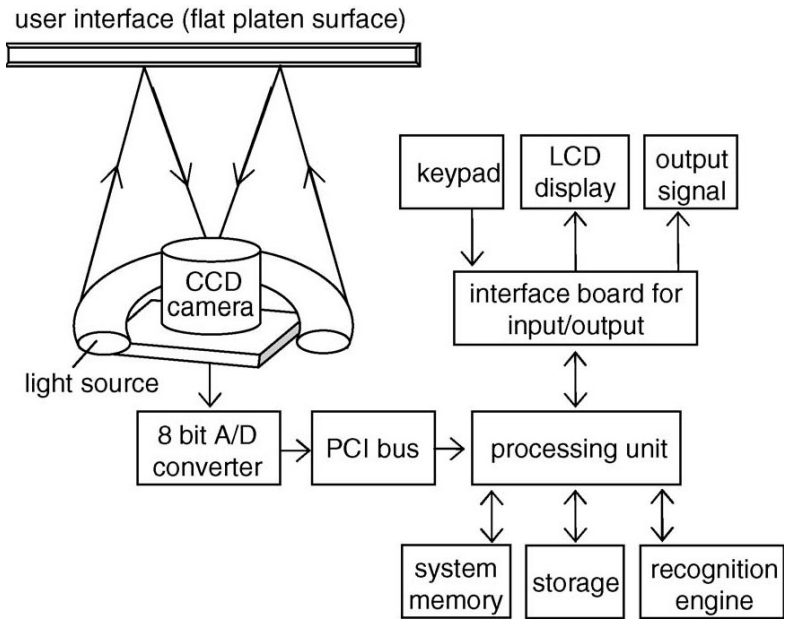

Figure 3.1 Schematic Diagram of Palmprint Acquisition System [7]

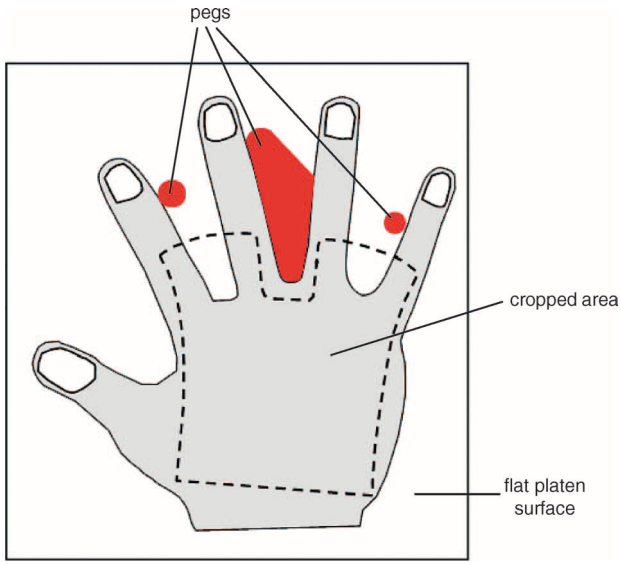

Figure 3.2 Pegs and the Cropped Area of the Palm [7]

Various algorithms have been developed to be used in palmprint recognition. Developed algorithms mainly include different methods for feature extraction and

distance matching. From now on, some of the methods developed for palmprint recognition will be mentioned and their results will be discussed.

Fang Li et al. [8] proposed an approach utilizing Line Edge Map (LEM) of palmprint as the feature and Hausdorff distance as the distance matching algorithm. In this study, Line segment Hausdorff distance (LHD) and Curve segment Hausdorff distance (CHD) are explored to match two sets of lines and two sets of curves. They carried out an identification experiment on The Hong Kong Polytechnic University Palmprint Database. 200 palm images, i.e. 2 palm images for each person, have been randomly selected in order to test the system performance. They reserved one palm image for each individual as a template, and used remaining palm images as test images to be identified.

Fang Li et al. [9] later proposed the utilization of Modified Line segment Hausdorff distance (MLHD) as the distance matching algorithm. In this study, 2-D lowpass filter is applied to sub-image extracted from the captured hand image. The result is subtracted from the image in order to decrease the non-uniform illumination effect resulting from the projection of a 3-D object onto a 2-D image. After line detection, contour and line segment generation steps, each line on a palm is represented using several straight line elements. Finally, MLHD is used in order to measure the similarity between two palm images. Performance of this and some other palmprint identification methods are tabulated in Table 3-1.

Table 3-1 Comparison of Different Palmprint Identification Methods

	Duta et al. [10]	**You at all [11]**	**Zhang et al. [12]**	**LHD [8]**	**CHD [8]**	**MLHD [9]**
Database Size	30	200	200	200	200	200
Feature	Feature Points	Texture and Feature Points	Lines	Lines	Curves	Lines
Matching Criteria	Euclidean Distance	Energy Difference and Hausdorff Distance	Euclidean Distance	Line Hausdorff Distance	Curve Hausdorff Distance	Modified Line Hausdorff Distance
Recognition Rates	% 95	% 91	% 92	% 96	% 92	%100

Algorithms employing neural networks have also been developed. Li Shang et al. [13] suggested the usage of radial basis probabilistic neural network (RBPNN). The RPBNN is trained by the orthogonal least square algorithm (OLS) and its structure is optimized by the recursive OLS algorithm (ROLSA). A fast fixed-point algorithm is used for independent component analysis. The Hong Kong Polytechnic University Palmprint Database is used to test the developed palmprint recognition algorithm. After tests performed on this database, recognition rates between % 95 and % 98 are obtained.

There are also methods utilizing morphological operations in order to extract features in a palmprint. Xiang-Qian Wu et al. [14] presented an approach based on valley features. Bothat operation, which utilizes two morphological operations; namely opening and closing, which are defined by combining dilation and erosion; is applied to extract the valleys in different directions in low-resolution images and to form the valley feature. After the valley feature has been obtained, a distance matching algorithm adopted to measure the similarity of the valley features has been

employed. ROC curve of this approach, which has an EER of % 2, and the resulting distributions of genuine-imposter matching scores are displayed in Figure 3.3. C. Han et al., on the other hand, utilized Sobel and morphological operations in order to enhance lines on a palm. They then divided palmprint into several sub-blocks and a feature vector has been obtained from the mean of pixel values in each sub-block. ROC curve of this method, which has an EER of % 14, is also shown in Figure 3.3.

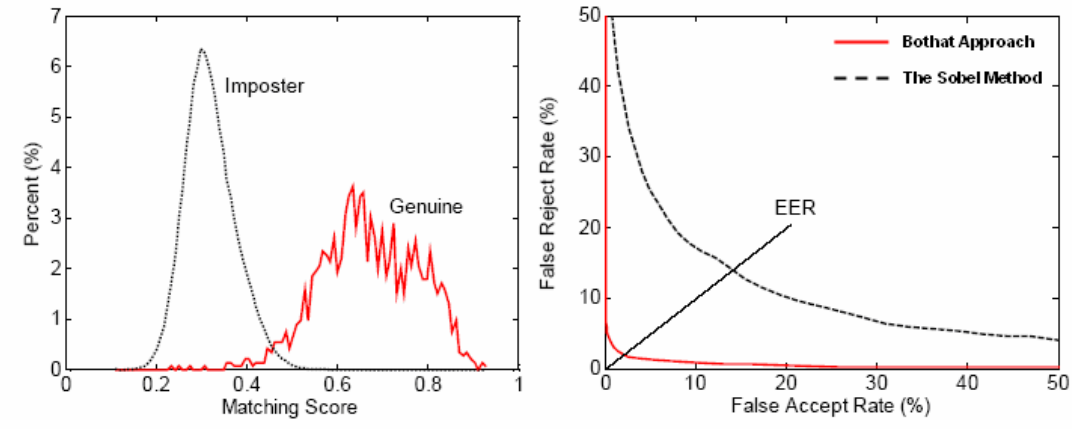

Figure 3.3 Distribution of Genuine and Imposter Matching Scores and ROC Curves [14]

Gabor filters, being widely used in fingerprint recognition algorithms [15-17], are also used in palmprint recognition algorithms. David Zhang et al. [6] proposed an approach utilizing 2-D Gabor filter in order to extract features. They used normalized Hamming distance to measure the similarity between palmprints. They achieved to reach an EER of % 0.6 in verification tests. ROC curve and the distributions of genuine-imposter matching scores obtained in verification tests are shown in Figure 3.4. They also performed identification tests. In this identification tests, they set up three databases consisting of 50, 100 and 200 individuals. They reserved 3 palm images of each user as templates resulting in 150, 300 and 600 templates, respectively. ROC curve obtained in identification tests for these three databases are shown in Figure 3.5. It is seen in Figure 3.5 that as the number of registered users increases, the identification accuracy of the system decreases. This is quite expected,

because number of total classes in a database is equal to the number of registered users, therefore it becomes harder for the system to classify correctly.

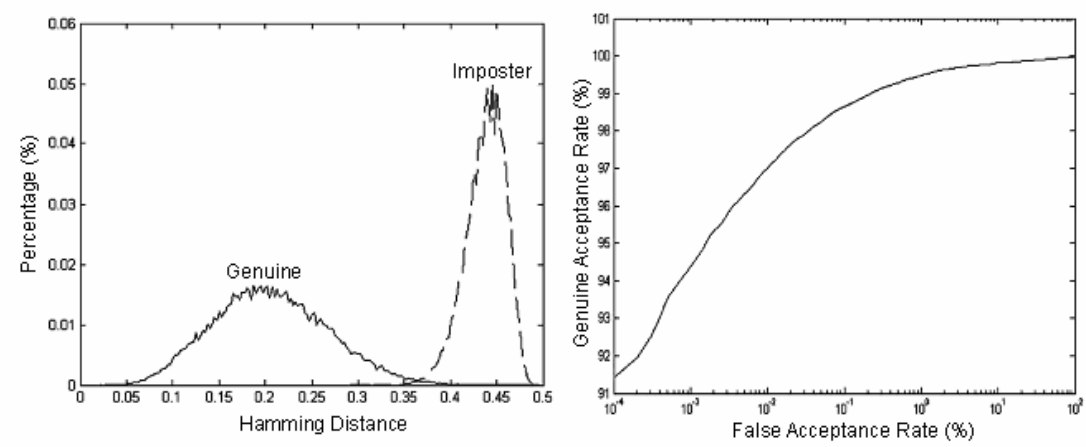

Figure 3.4 Distribution of Genuine-Imposter Matching Scores and ROC Curve – Verification [6]

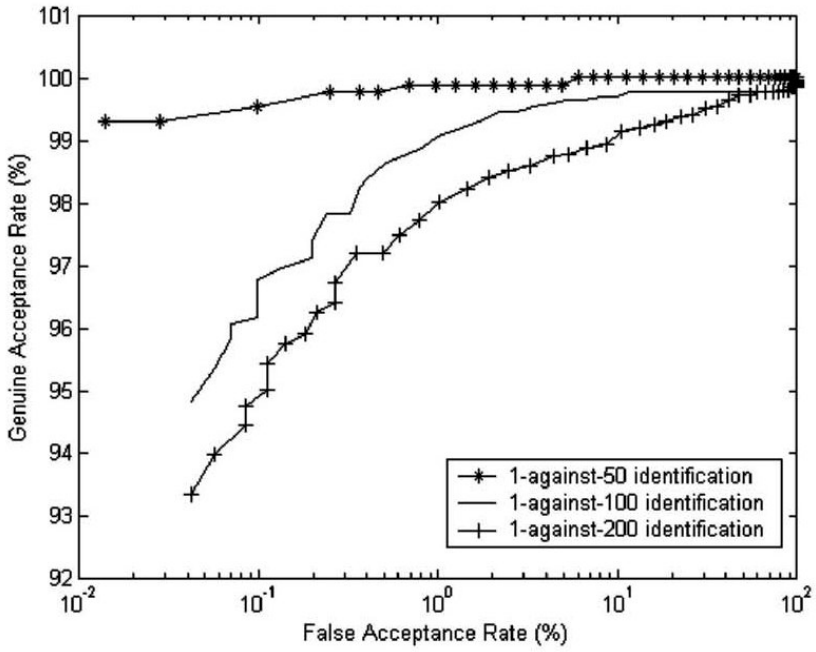

Figure 3.5 ROC Curve – Identification [6]

Another algorithm utilizing Gabor filters is studied by Xiangqian Wu et al. [18]. This algorithm is based on the fusion of the phase information (called FusionCode) and the orientation information (called OrientationCode). In this study, 4 Gabor filters with different orientations are used to extract the FusionCode and the OrientationCode. After the FusionCode and the OrientationCode have been obtained, they are fused to obtain the feature vector, Palmprint Phase Orientation Code. Finally, modified Hamming distance is used to measure the similarity between two palm images. In verification tests, they reached an EER of 0.3 %. The distribution of genuine-imposter matching scores for this approach is shown in Figure 3.6. In order to see the effect of the fusion on verification accuracy, they also implemented Fusion Code method and the OrientationCode technique. Resulting ROC curves of these three methods are also shown in Figure 3.6.

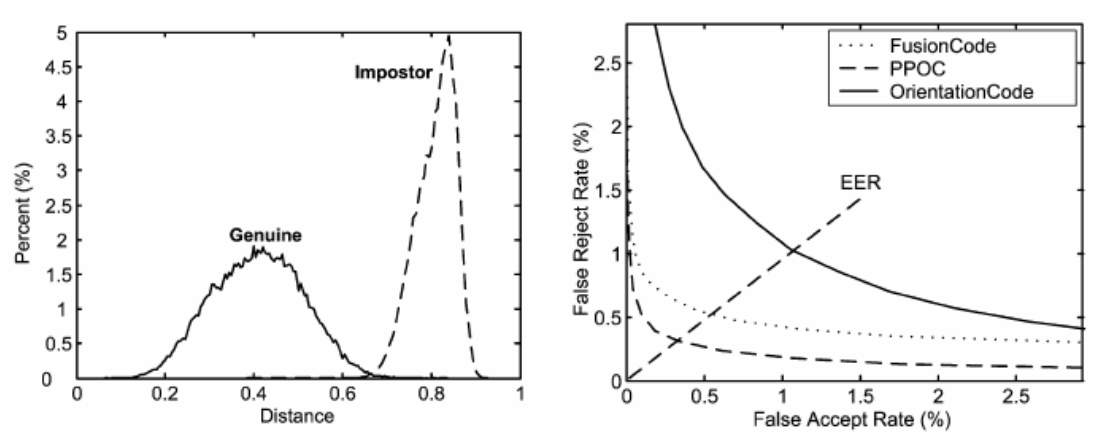

Figure 3.6 Distribution of Genuine-Imposter Matching Scores and ROC Curves [18]

Ajay Kumar and Helen C. Shen [19] also proposed an approach in which Gabor filter is used. In this approach, 800 images have been acquired from 40 individuals, 10 from their right palms and 10 from their left palms, via HP-Scanjet ADF scanner. Acquired palm images are first normalized in order to reduce brightness and contrast variations resulting from sensor noise and variations in palm pressure. The normalization process is depicted in Figure 3.7. After normalization, each image is subjected to multi-channel filtering using a bank of Real Gabor Function (RGF) filters. These filtered images, shown in Figure 3.8, are used to extract features from

each of 6 concentric circular bands. Finally, the similarity between feature vectors is measured and each palm image is classified into a class. Experimental results show that they reached an EER of 3.03 % when total number of classes is considered to be 80, that is, left and right palms of each individual are counted as two different classes. This result proves the uniqueness of palmprint texture even in two hands of an individual.

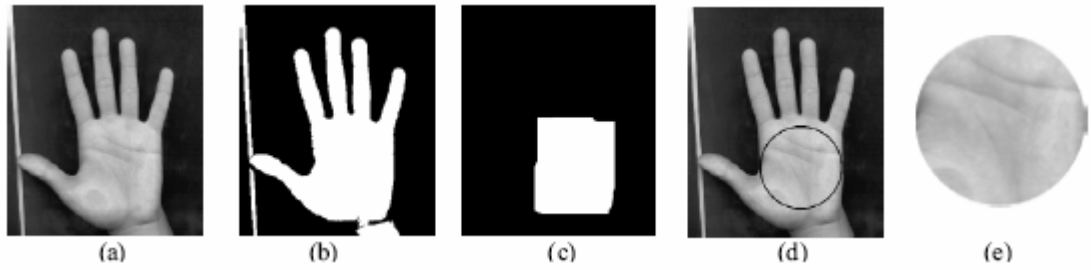

Figure 3.7 (a) Acquired Palm Image, (b) Palm Image After Thresholding, (c) Image Residue After Morphological Erosion, (d) Circular Region of Interest, (e) Segmented Palmprint Image [19]

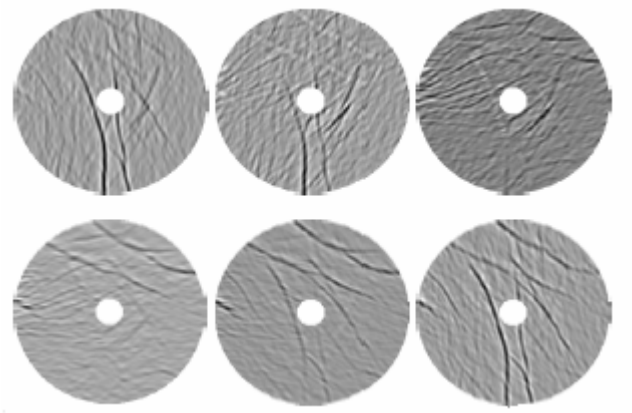

Figure 3.8 Images Filtered by Six Different Real Gabor Function Filters [19]

As it is seen, having proven its accuracy in many different algorithms, palmprint-scan technology stems as a new reliable biometric technology. Details of the proposed algorithm which is based on 2-D Gabor filters, is given in the next chapter.

CHAPTER IV

PROPOSED ALGORITHM

This chapter details the proposed algorithm for palmprint recognition. The Hong Kong Polytechnic University Palmprint Database has been used during the development of the palmprint recognition algorithm. Moreover, the developed algorithm is first tested on this database. The block diagram of the proposed algorithm is shown in Figure 4.1. The organization of this chapter is as follows: Section 4.1 details preprocessing performed on palmprint images in the database. Section 4.2 describes the feature extraction and the coding block. Finally, Section 4.3 briefly explains the distance matching algorithm and the decision policy.

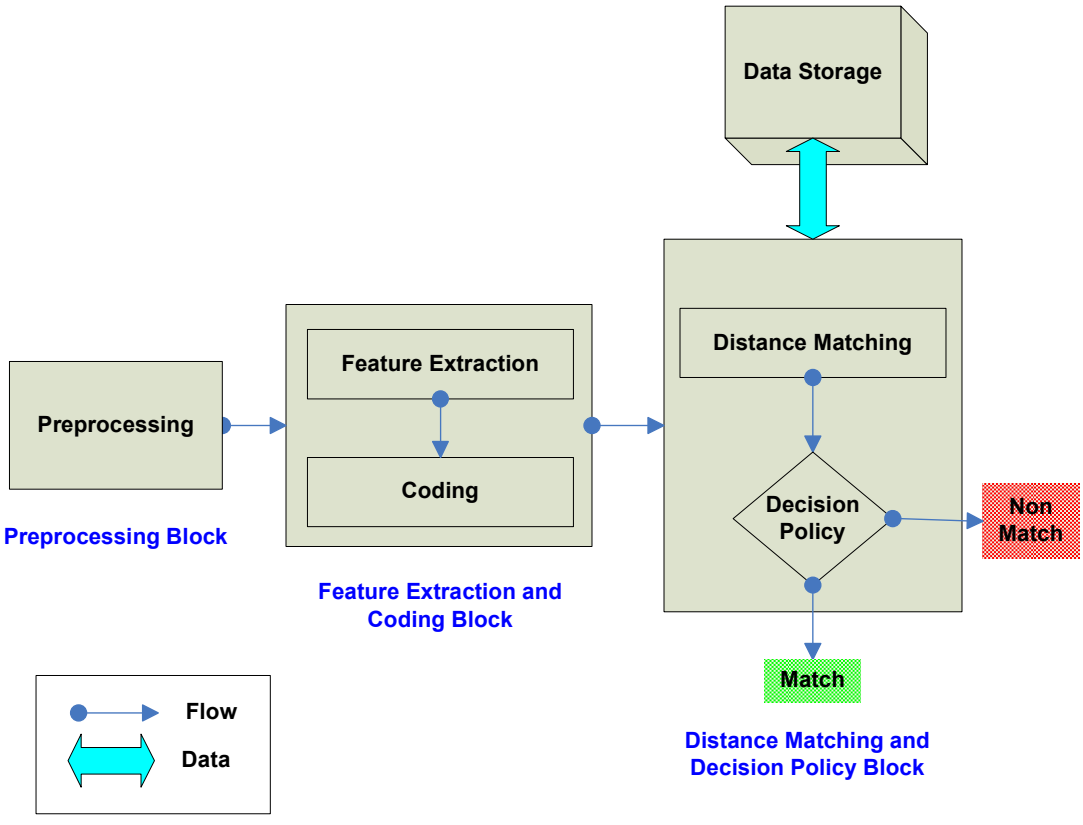

Figure 4.1 Block Diagram of the Proposed Algorithm

4.1 Preprocessing

The preprocessing block, which is the first block in the developed algorithm as it is the case in many biometric systems, is one of the most critical parts of the developed palmprint recognition algorithm. Before feature extraction and coding step, all images in the database should be preprocessed and the central area of a palm should be obtained. Preprocessing algorithm that will be used for this purpose should be selected such that the algorithm should be applicable to all images in the database and the desired area of the palm should be obtained with great accuracy. Otherwise, extracted features will not belong to the desired part of the palm; therefore accuracy of the developed palmprint recognition algorithm would significantly decrease. In brief, accuracy of the preprocessing algorithm is very important, since possible errors in this block will affect subsequent blocks.

Having inspected all images in the database, we decided to define the coordinate system as given in Figure 4.2 to align different palm images. Point A and Point B shown in Figure 4.2 represent edge points to be found in preprocessing and the vertical black line passing through both Point A and Point B form the Y-axis. The horizontal black line is perpendicular to the Y-axis, hence it forms the X-axis. Nevertheless, one can draw infinitely many lines which are perpendicular to the Y-axis, unless their intersection point, I, is specified. Therefore; the intersection point I has to be specified in order to have unique X and Y axes. Not surprisingly, the midpoint of A and B is selected as the intersection point, I. Uniqueness of X and Y axes can be proven as follows: Since only one line can pass through two different points, A and B in our case, Y-axis is unique. Since the Y-axis is obtained, its slope is known and because the X-axis is perpendicular to the Y-axis, the slope of the X-axis is also known. It is above stated that, the midpoint of A and B is selected as the intersection point, I, hence the X-axis passes through I. Therefore; one point on the X-axis and the slope of the X-axis are known; hence the equation of the X-axis is known and it is unique.

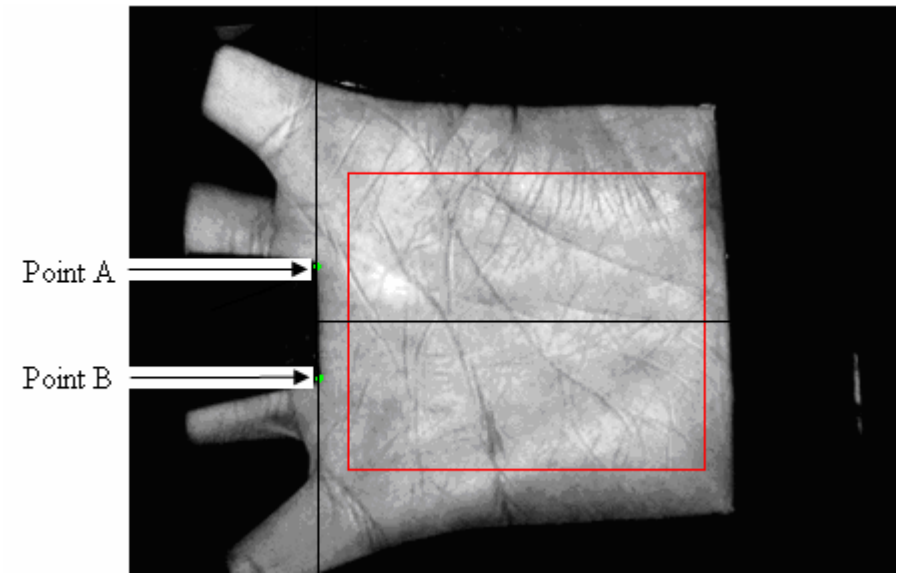

Figure 4.2 The Coordinate System

It is obvious that two reference points, A and B, should be first obtained in order to determine the coordinate system mentioned above. In order to obtain these reference points, a cropped image similar to the one shown in Figure 4.3 will be sufficient.

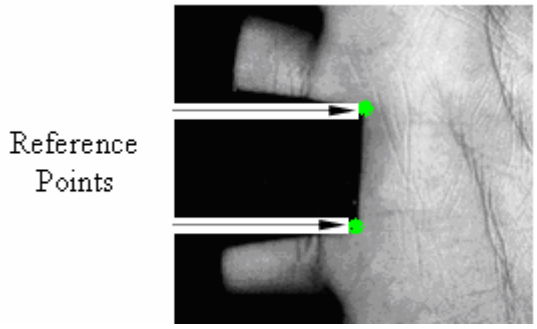

Figure 4.3 A Cropped Palm Image Including Reference Points

After the palm image is cropped, the Canny edge detector in MATLAB® is used and the resulting image is rotated 90° in the clockwise direction. Hence the image shown in Figure 4.4 is obtained.

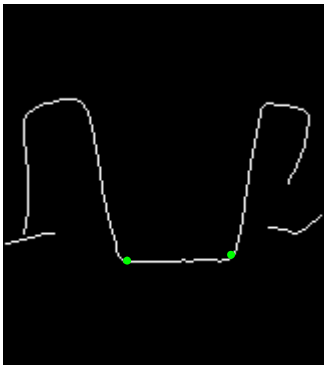

Figure 4.4 After Canny Edge Detector and 90° Clockwise Rotation

As it is seen in Figure 4.4, the boundary of the desired region is obtained after these operations. The next step is to find the reference points, A and B, which are on this boundary. Nonetheless, it is here worth noting that, even very small errors in locations of the reference points may significantly affect the coordinate system, because the reference points determine both the origin of the coordinate system, that is I, and the slopes of X and Y axes. Let the upper left corner of Figure 4.4 be the origin denoted by O (0, 0) and let any point p in Figure 4.4 be denoted as (x (p), y (p)), where x (p) is the horizontal distance of point p to the origin in pixels and y (p) is the vertical distance of the same point to the origin in pixels. Traversing the boundary shown in Figure 4.4 from left to right, (x (p), y (p)) values change as shown in Figure 4.5.

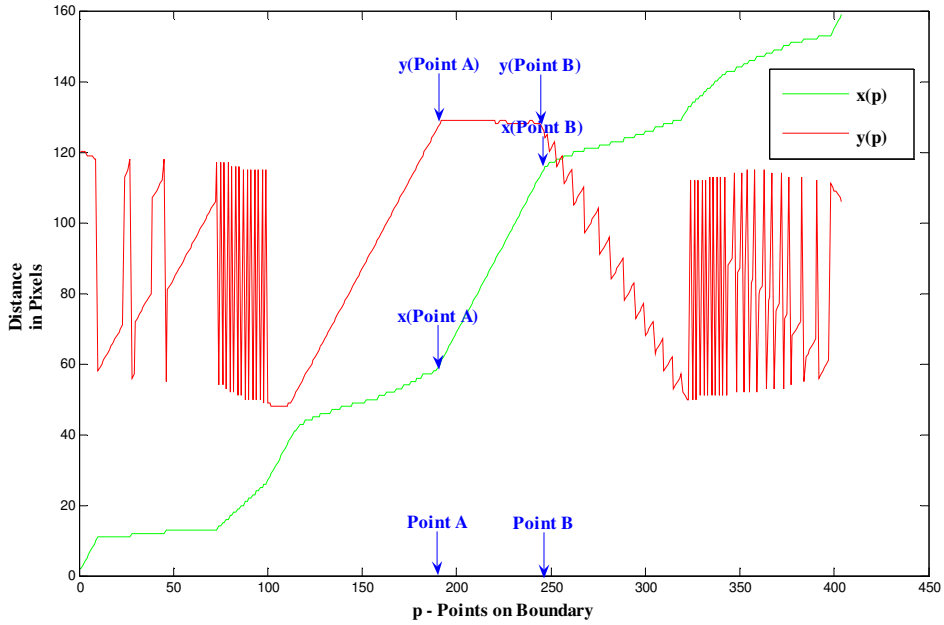

Figure 4.5 Changes in Vertical and Horizontal Distances

The reference points are roughly shown in Figure 4.5. As it can be seen the vertical distance of a point p to the origin O, y (p), is nearly constant between reference points, whereas the horizontal distance of the same point to the origin O, x (p), increases very rapidly in the same interval. It is quite expected because in Figure 4.4 it is seen that when we traverse the boundary between reference points from left to right, we move horizontally; therefore x (p) increases linearly and y (p) remains nearly constant. It is also seen in Figure 4.5 that the increase in x (p) values is the fastest between reference points. That's why when the derivate of x (p), which is just the difference of two consecutive elements x (n) and x (n - 1) since x (p) is a discrete function of p, is filtered with a rectangular window of length N, rough values of reference points can be obtained. This is because the value of the resulting function between reference points will be at its maximum value and will also be very close to N. This is shown in Figure 4.6, where N is empirically set to 40.

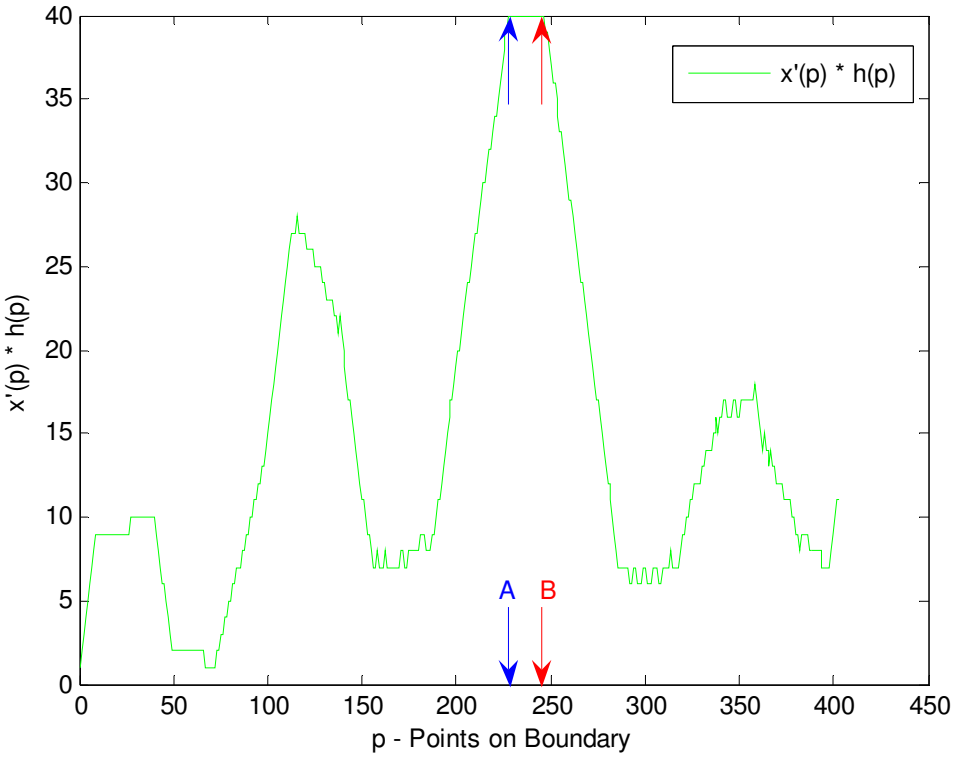

Figure 4.6 After Filtering x'(p) with a Rectangular Window

As it is stated above, y (p) remains nearly constant between reference points and this information can be used to correct the location of reference points, if necessary. In order to find exact locations of the reference points, rough values of the reference points shown in Figure 4.6 are used. The midpoint of these points is taken as the starting point and the algorithm searches for points where y (p) starts to change in both directions. However, there may be slight variations in y (p) in both directions and the algorithm should not interpret these variations as y (p) changes. This is performed by filtering the derivate of y (p) with a rectangular window of length 7 and comparing the resulting function to a threshold. Rough values of reference points found after filtering the derivate of x (p) are shown in red in Figure 4.7. Corrected reference points obtained after filtering the derivative of y (p) are shown in blue in the same figure.

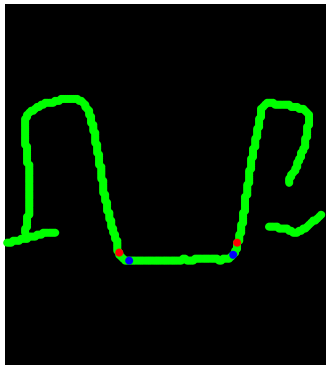

Figure 4.7 Rough Values of Reference Points (Red) and Corrected Reference Points (Blue)

As it is already stated, locations of reference points are enough to form the coordinate system shown in Figure 4.2. Rectangular region shown in Figure 4.2 is the desired palm area and vertical sides of this rectangular region are parallel to the Y-axis. That is why; when the slope angle of the Y-axis is different from 90°, the slope angle of the rectangle's vertical sides is also different from 90°. It is here worth noting that pixel values in an image sit on a rectangular grid and when the slope angle of the vertical sides is different from 90°, extracted rectangular region does not fit a rectangular grid. The easiest solution to this problem is that the palm image should be rotated by an angle of θ° in the clockwise direction if the slope angle of the Y-axis is (90 + θ) ° and it should be rotated by an angle of θ° in the counter-clockwise direction if the slope angle of the Y-axis is (90 - θ) °. When all images in the database are inspected, it is seen that the maximum value of θ will be around 7°. Rotation of an image with an angle different from (n * 90) °, where n is an integer, needs interpolation and interpolation degrades the quality of an image. However, since the rotation angle is quite small and all images in the database are subject to rotation, i.e. all images in the database are affected in somewhat same manner; degradation in quality of images caused by rotation is negligible. After rotation, the Y-axis will have a slope angle of 90°; hence the extracted rectangular region will fit the rectangular grid. Let reference points be at A (x_A, y_A) and B (x_B, y_B), then the rotation angle θ in counter-clockwise direction can be calculated as follows:

$$\theta = \left(90 - \tan^{-1}\left(\frac{y_B - y_A}{x_B - x_A} \right) \right), \text{ in degrees} \quad (4.1)$$

After palm images are rotated by an angle of θ°, same operations are performed to locate reference points on rotated images. Figure 4.8, Figure 4.9, Figure 4.10 and Figure 4.11 show reference points located on the same palm image before and after rotation.

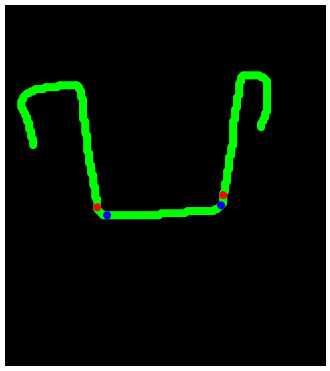

Figure 4.8 Reference Points on a Cropped Palm Image Before Rotation

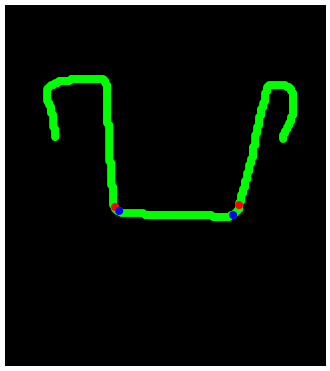

Figure 4.9 Reference Points on the Same Cropped Palm Image After Rotation

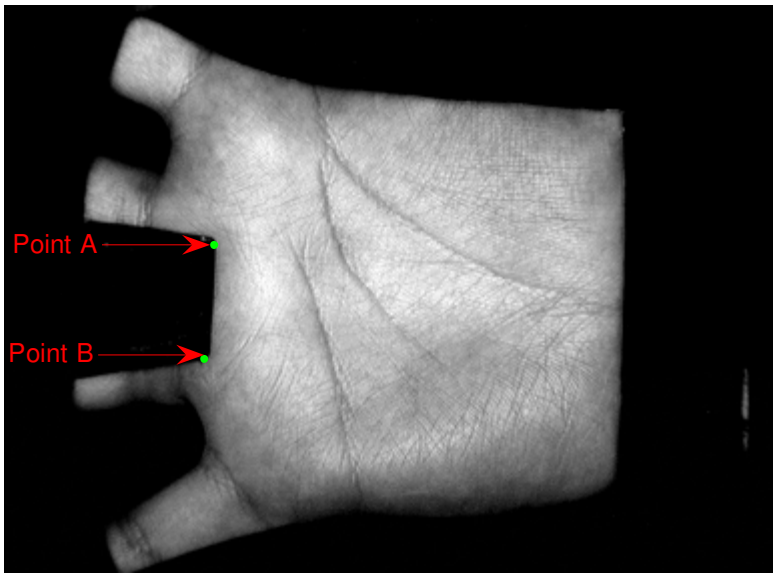

Figure 4.10 Reference Points on a Palm Image Before Rotation

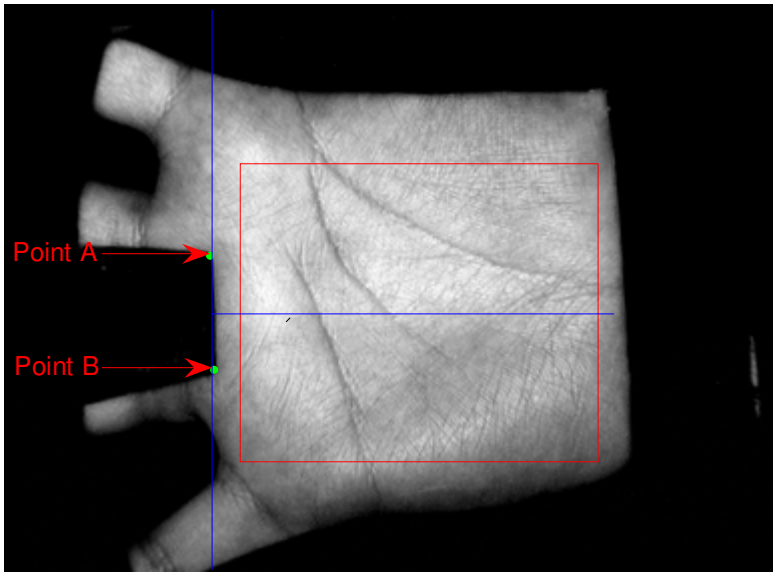

Figure 4.11 Reference Points on the Same Palm Image After Rotation

After reference points are located on rotated palm images, the next step is to extract the central palm area, which is the rectangular region shown in Figure 4.11. After rotation, the slope angle of the Y-axis is 90° therefore; the slope angle of the rectangular region's vertical sides is also 90°. This results in that the extracted rectangular region fits on the rectangular grid. Since X-axis is perpendicular to the Y-axis, the slope angle of the X-axis is 0°. Let reference points be at A (x_A, y_A), B(x_B, y_B) and noting that $x_A = x_B$ after rotation, then the coordinates of the midpoint of A and B, (x_o, y_o), can be found as follows:

$$\begin{aligned} x_o &= x_A = x_B \\ y_o &= \frac{y_A + y_B}{2} \end{aligned} \tag{4.2}$$

The rectangular region has a length of 180 pixels in horizontal direction and a width of 150 pixels in vertical direction. Also the perpendicular distance between the left vertical side of the rectangular region and the Y-axis is 15 pixels. Let $R_1(x_1, y_1)$ denote the coordinates of the upper left corner of the rectangular region, and similarly, $R_2(x_2, y_2)$ denote the coordinates of the lower left corner of the rectangular region, then:

$$\begin{aligned} x_1 &= x_o + 15 \\ y_1 &= y_o + 75 \end{aligned} \tag{4.3}$$

$$\begin{aligned} x_2 &= x_o + 15 \\ y_2 &= y_o - 75 \end{aligned} \tag{4.4}$$

After coordinates of the lower and upper left corners are evaluated, coordinates of the upper right corner, $R_3(x_3, y_3)$, and the lower right corner, $R_4(x_4, y_4)$ can be obtained as follows:

$$\begin{aligned} x_3 &= x_1 + 180 \\ y_3 &= y_1 \end{aligned} \tag{4.5}$$

$$x_4 = x_2 + 180$$
$$y_4 = y_2 \tag{4.6}$$

Because all coordinates of the rectangular region have been determined, the desired central palm area is ready to be extracted. Figure 4.12 shows a sample palm image and Figure 4.13 shows the extracted central palm area for the same palm.

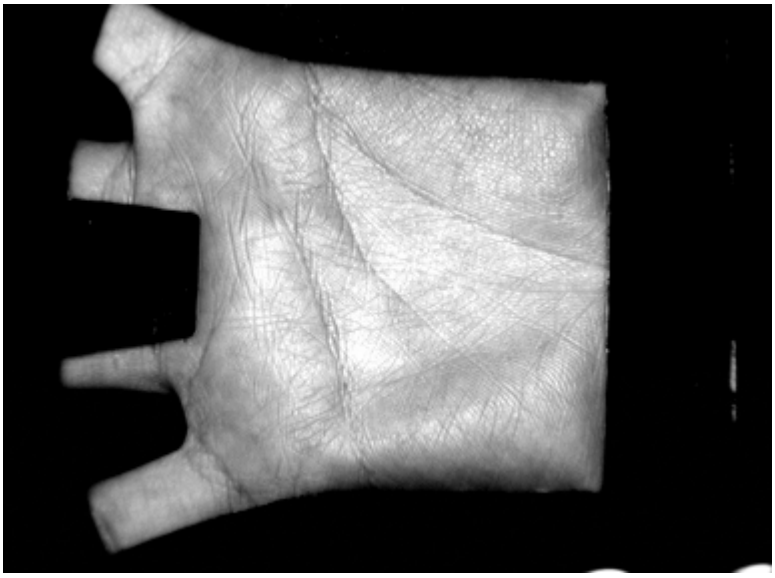

Figure 4.12 Sample Palm Image

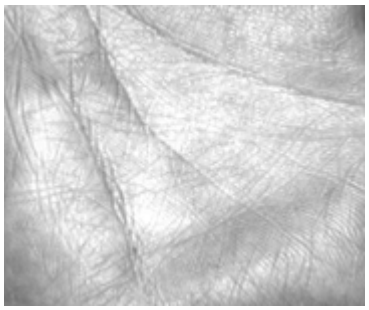

Figure 4.13 Extracted Region for the Palm Image in Figure 4.12

Since the desired area of the palm is extracted, preprocessing is complete. The next block, namely the feature extraction and coding block, is detailed in the next section.

4.2 Feature Extraction and Coding

In this block, relevant features are extracted from the central palm area obtained in the previous block. Then these extracted features are coded and the mathematical representation of the palm is obtained.

Developing a palmprint recognition algorithm that can successfully discriminate between palm images of low resolution is a big advantage from the practical side of view. This is because; since the developed algorithm does not require high resolution images, there is no need in high resolution capturing devices which are quite expensive. Being aware of the fact that the cost of the capturing device plays an important role in determining the total cost of the developed biometric system, it can be said that the total cost of the system can be significantly decreased by decreasing the cost of the capturing device. It should be obvious that low-cost products are easy to market therefore; developing an algorithm capable of working accurately with low resolution images is very important.

Principal lines, wrinkles, ridges, minutiae points and texture are considered to be relevant features for a palm (Three principal lines, named as Life Line, Heart Line and Head Line, and some wrinkles in a palm are shown in Figure 4.14). However, these relevant features require different resolutions in order to be extracted. In general, principal lines and wrinkles can be extracted from low resolution images, whereas ridges and minutiae points need higher resolution. Table 4-1 shows approximate required resolutions to extract principle lines, wrinkles and ridges texture in dots per inch (dpi). As it is seen from the table, principal lines can be obtained even in quite low resolution images. Considering the cost of the biometric system, principal lines may be thought to be very suitable to be used in the developed algorithm. Although principal lines can be extracted with algorithms such as the stack filter, they do not have the uniqueness property, that is, different individuals

may have similar principle lines. This problem has been demonstrated in Figure 4.15. Palm images in (a), (b) and (c); (d), (e) and (f); and (g), (h) and (i) are very similar to each other; however they belong to different individuals. Wrinkles may also be thought to be employed, nevertheless; usage of wrinkles is questionable due to the permanence property, because wrinkles are subject to change with time. Furthermore, extracting wrinkles accurately is not an easy task. Due to reasons mentioned above, texture analysis has been selected to be used in the developed algorithm. [6]

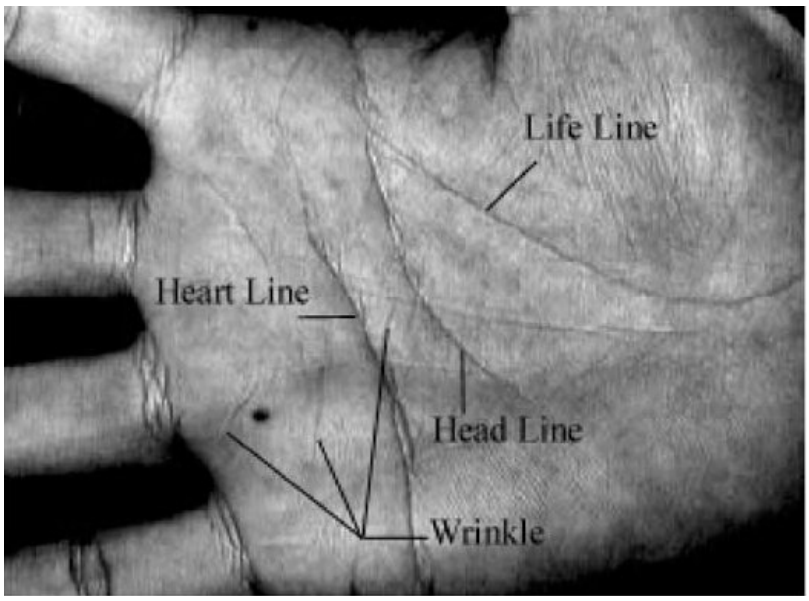

Figure 4.14 Principle Lines and Wrinkles in a Palm [20]

Table 4-1 Resolution Requirements for Different Palmprint Features

Palmprint Features	Required Resolution (in dpi)
Principal Lines	≥ 75
Wrinkles	≥ 100
Ridges texture	≥ 125

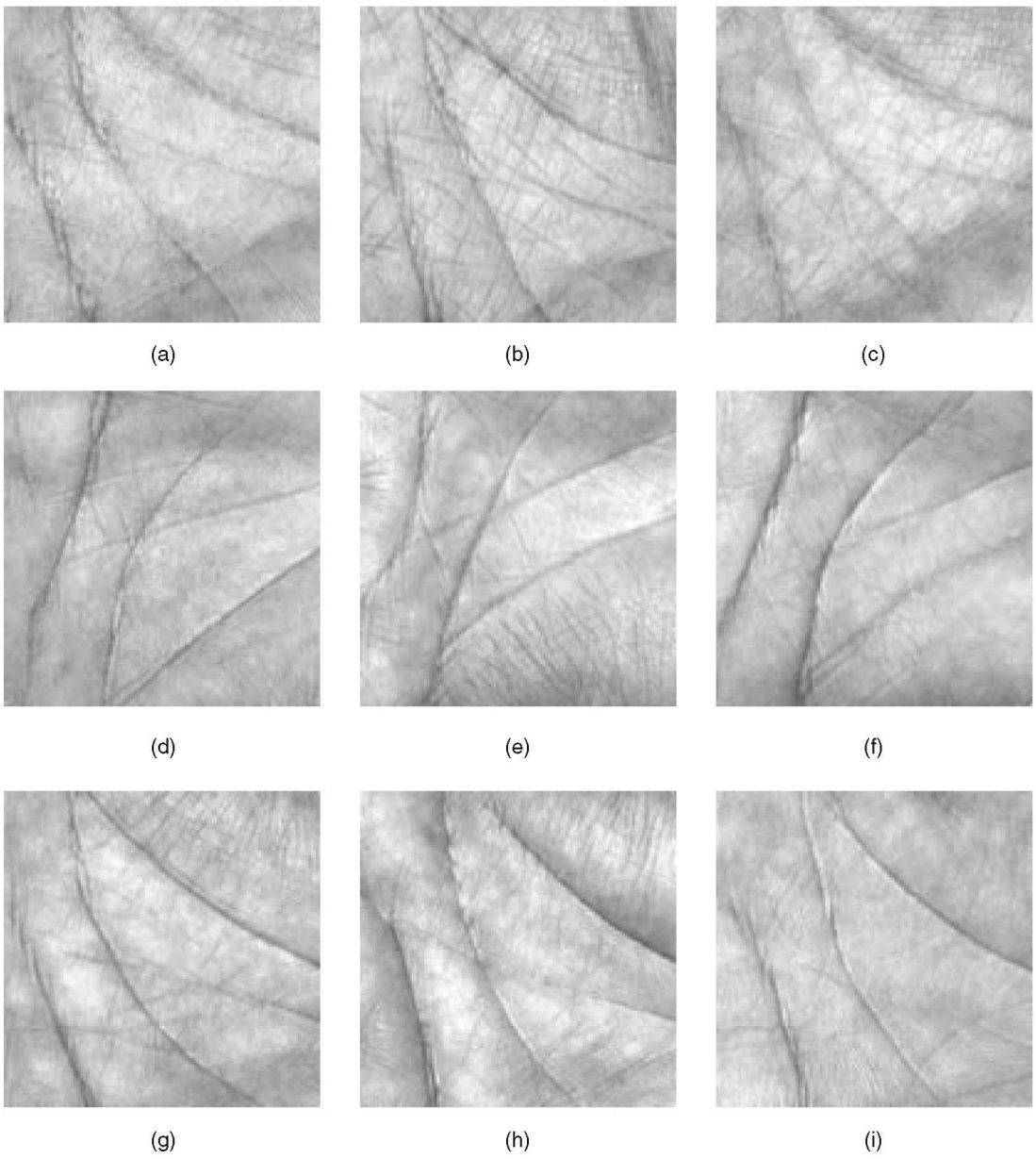

Figure 4.15 Three Sets of Palmprint Images with Similar Principal Lines from Different People [6]

Having proven its success in fingerprint and face recognition, Gabor filters are chosen for texture analysis [15, 16, 17, 21, 22, 23]. Before going into details of the developed algorithm, it is worth giving a short summary of Gabor filters.

4.2.1 Gabor Filter

Having modeled the spatial summation properties of the receptive fields in the primary visual cortex, Gabor filters are very popular in various image processing applications. [24] A Gabor filter is a linear filter whose impulse response is defined by a sinusoidal function multiplied by a Gaussian function. [25] Basically, Gabor filters can be considered as a sinusoidal plane modulated by a Gaussian envelope. [26] Figure 4.16 demonstrates a sample one-dimensional Gabor filter and its Gaussian and sinusoidal components. Fourier transforms of 1-D Gabor Filter and its components are shown in Figure 4.17.

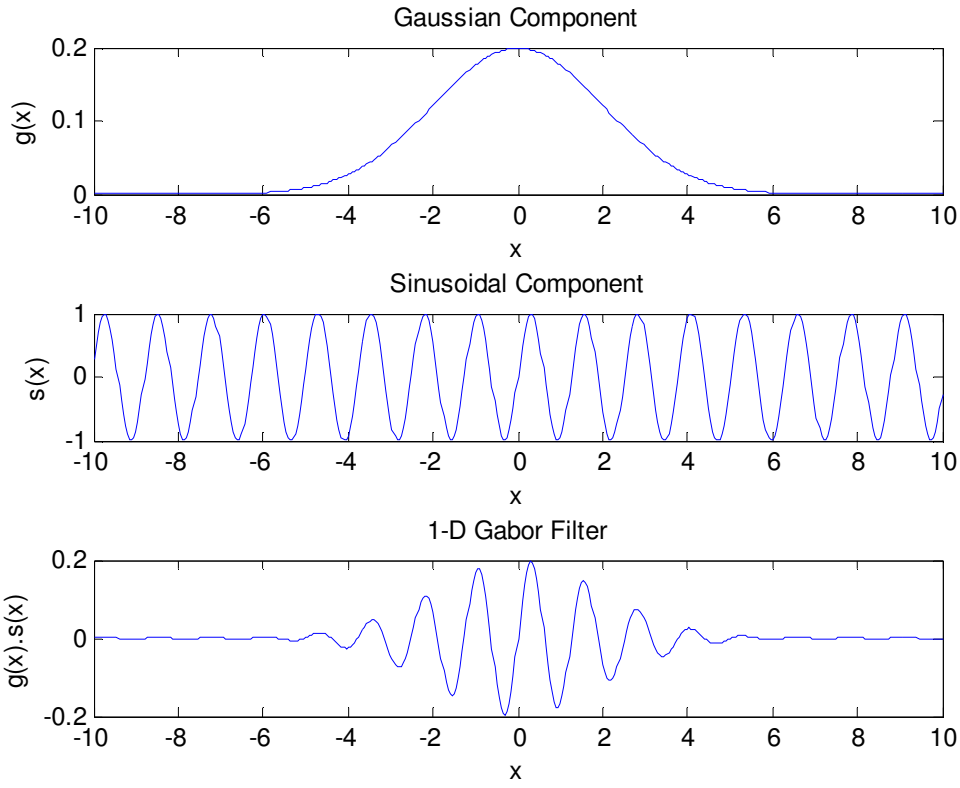

Figure 4.16 1-D Gabor Filter and Its Components

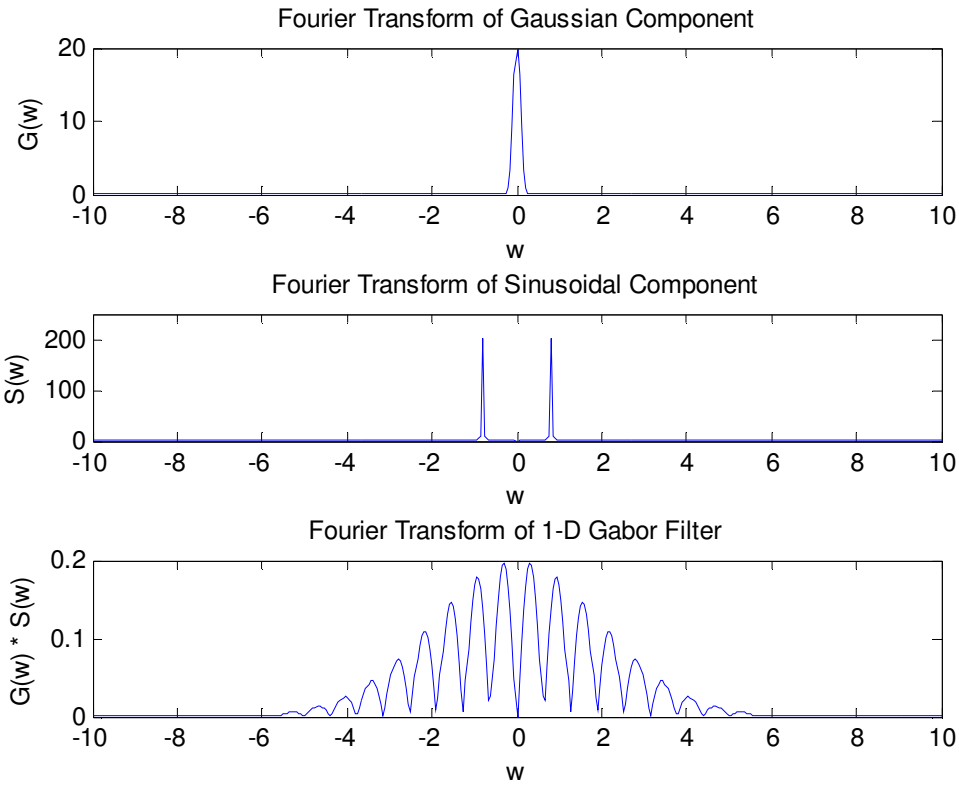

Figure 4.17 Fourier Transforms of 1-D Gabor Filter and Its Components

Two-dimensional (2-D) Gabor filters were originally proposed in 1980 by Daugman as a framework for understanding the orientation-selective and spatial-frequency-selective receptive field properties of neurons in the brain's visual cortex, and as useful operators for practical image analysis problems. Their mathematical properties were further elaborated by the author in 1985, who pointed out that these 2-D quadrature phasor filters were conjointly optimal in providing the maximum possible resolution both for information about the orientation and spatial frequency content of local image structure (in a sense “what”), simultaneously with information about 2-D position (in a sense “where”). In other words, Gabor filters are extremely useful for texture analysis because of the 2-D spectral specificity of texture as well as its variation with 2-D spatial position. In addition to accurate time-frequency location, they also provide robustness against varying brightness and contrast of images. [27]

2-D Gabor filter has the following general form:

$$G_{\gamma,\sigma,\lambda,\varphi,\theta}(x,y) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right)\exp\left(2\pi j\left(\frac{x'}{\lambda} + \varphi\right)\right) \tag{4.7}$$

where

$$\begin{aligned} x' &= x\cos(\theta) + y\sin(\theta) \\ y' &= -x\sin(\theta) + y\cos(\theta) \end{aligned} \tag{4.8}$$

Parameters involved in Equations (4.7) and (4.8) are [24] [26]:

- γ, the spatial aspect ratio, specifies the ellipticity of the Gabor function. When γ = 1, the filter is called circular Gabor filter.
- σ, the standard deviation of the Gaussian factor, determines the linear size of the receptive field.
- λ, the wavelength of the sinusoidal function, specifies the spatial frequency of the cosine factor.
- φ, the phase offset in the argument of the cosine factor, determines the symmetry of the concerned Gabor function. Real part of the function is even for φ = 0° and φ = 180°; and the function is odd for φ = ±90°.
- θ, the angle parameter, specifies the orientation of the normal to the parallel stripes of the Gabor function.
- x and y are spatial variables.

After replacing x', y' and inserting γ = 1 and φ = 0, Equation (4.7) can be written as:

$$G^{R}{}_{\sigma,\lambda,\theta}(x,y)=\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)\cos\left(2\pi\left(\frac{x\cos\theta+y\sin\theta}{\lambda}\right)\right)$$

$$G^{I}{}_{\sigma,\lambda,\theta}(x,y)=\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)\sin\left(2\pi\left(\frac{x\cos\theta+y\sin\theta}{\lambda}\right)\right) \quad (4.9)$$

$$G_{\sigma,\lambda,\theta}(x,y)=G^{R}{}_{\sigma,\lambda,\theta}(x,y)+\mathrm{j}\,G^{I}{}_{\sigma,\lambda,\theta}(x,y)$$

where $G^{R}{}_{\sigma,\lambda,\theta}(x,y)$ and $G^{I}{}_{\sigma,\lambda,\theta}(x,y)$ represent the real and imaginary parts of the Gabor filter, respectively.

4.2.2 Gabor Filter Utilized in the Proposed Algorithm

In order to increase the robustness of the system against brightness, a zero-mean Gabor filter is preferred. In Equation (4.9), it is seen that the mean of the imaginary part of 2-D Gabor filter is zero due to odd-symmetry of the sine function. Nevertheless, the mean of the real part of the filter is non-zero because of the even symmetry of the cosine function. A zero-mean Gabor filter, $G^{ZM}{}_{\sigma,\lambda,\theta}(x, y)$, can be obtained by subtracting the mean of the Gabor filter from itself as shown in Equation (4.10):

$$G^{ZM}{}_{\sigma,\lambda,\theta}(x,y)=G_{\sigma,\lambda,\theta}(x,y)-\frac{\sum_{i=-N}^{N}\sum_{j=-N}^{N}G_{\sigma,\lambda,\theta}(i,j)}{(2N+1)^2} \quad (4.10)$$

where the filter size is (2N+1) by (2N+1).

The success of the developed algorithm highly depends on the Gabor filter parameters; namely σ, λ, and θ. The selection of these parameters affects not only the accuracy of the developed algorithm, but also the size of templates to be stored in the database. Furthermore, the size of the selected Gabor filter directly affects the speed of the algorithm. Effects of the filter parameters and the filter size on accuracy, speed and template size have been taken into account and these parameters and the size are tuned. Finally, filter parameters are determined to be σ = 1.4045, λ = 5.4555 and θ =

90°. Also the size of the Gabor filter is selected as 9x9. The magnitude of the resulting Gabor filter is shown in Figure 4.18. In addition, the magnitude of the Fourier Transform of the resulting Gabor Filter is demonstrated in Figure 4.19.

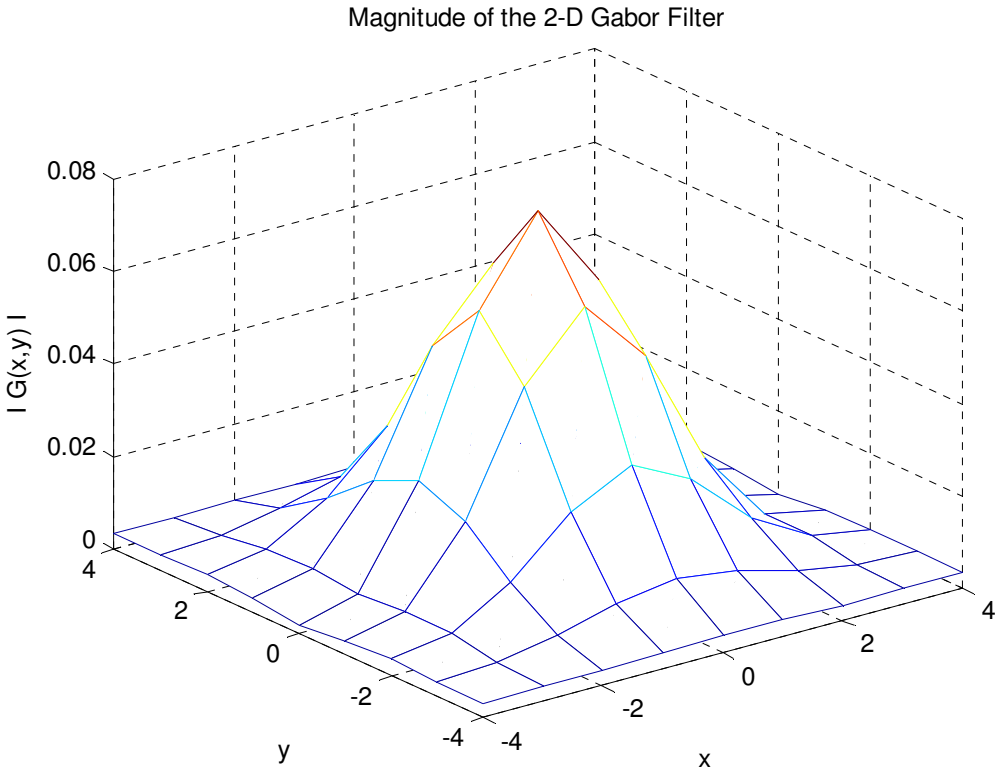

Figure 4.18 The Magnitude of the 2-D Gabor Filter

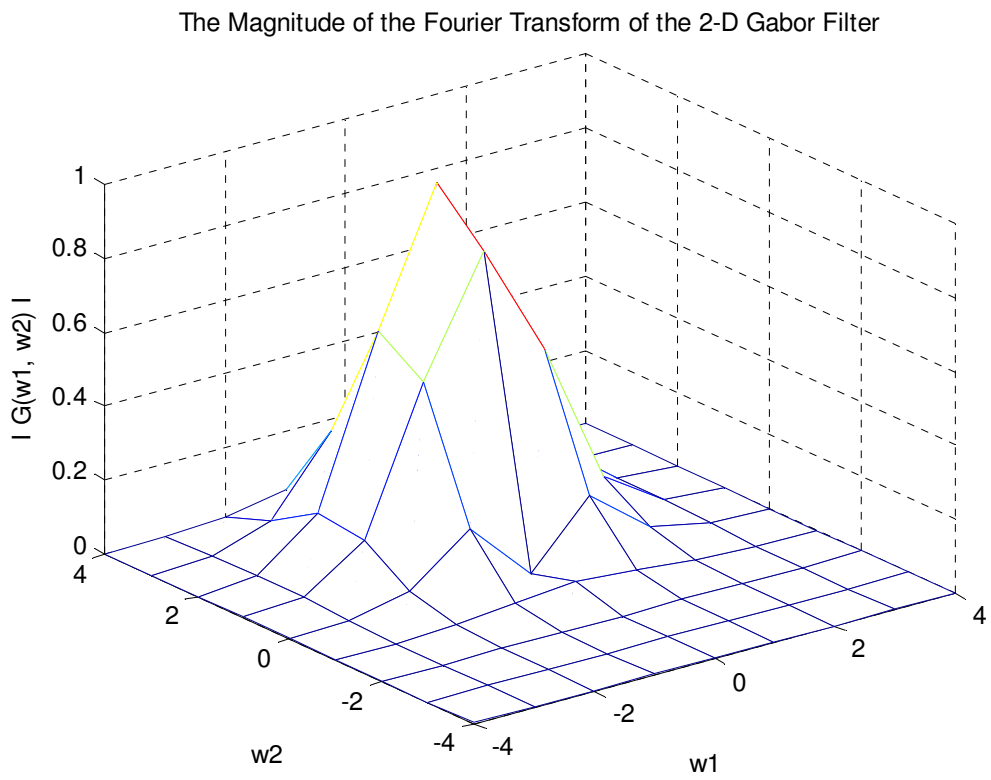

Figure 4.19 The Magnitude of the Fourier Transform of the 2-D Gabor Filter

Since the size and the parameters of the Gabor filter have been determined, Gabor filter coefficients can be easily calculated. Moreover, since the size and the parameters are constant, Gabor filter coefficients are also constant. This means that the Gabor filter coefficients can be calculated offline, which speeds up the developed algorithm by eliminating the need in recalculating the filter coefficients.

After the Gabor filter has been determined, the extracted region of the palm, obtained in the preprocessing block, is convoluted with the Gabor filter. Since the Gabor filter is a complex filter, the result of the convolution is also complex, having real and imaginary parts. After convolution, both the real and imaginary parts of the resulting image are divided into sub-blocks of 3x3. In these sub-blocks, the mean value of corresponding 9 pixels has been calculated. After the mean values of each sub-block have been calculated, mean values are compared to a threshold of -0.2. Then each sub-block is encoded as 1 if the mean value of the corresponding sub-block is bigger

than or equal to the threshold and encoded as 0 otherwise. Mathematically expressing the encoding:

$$b_{Real}(k,l) = u\left(\frac{\sum_{i=3k}^{3k+2}\sum_{j=3l}^{3l+2} I_{Real}(i,j)}{9} - Threshold\right)$$
$$b_{Imag}(k,l) = u\left(\frac{\sum_{i=3k}^{3k+2}\sum_{j=3l}^{3l+2} I_{Imag}(i,j)}{9} - Threshold\right) \quad (4.11)$$

where $I_{Real}(i, j)$ and $I_{Imag}(i, j)$ stand for the convolution results of real and imaginary parts at pixel (i, j), $b_{Real}(k, l)$ and $b_{Imag}(k, l)$ represent the encoding results of the real and imaginary parts of sub-block(k, l), and u() and Threshold stand for the unit-step response and the selected threshold, -0.2, respectively. As it is mentioned above, $b_{Real}(k, l)$ and $b_{Imag}(k, l)$, the matrices b_{Real} and b_{Imag} will be called as feature vectors from now on, can either be 1 or 0 for all (k, l), therefore; 1-bit is enough to represent the encoding result. This 1-bit encoding also decreases the sizes of the templates. It is here worth noting that, since the size of the extracted region is 180x150, the size of the encoding result is 60x50. Figure 4.20 shows an extracted region of a sample palm image and Figure 4.21 shows the real and the imaginary parts of the encoding result for the image in Figure 4.20.

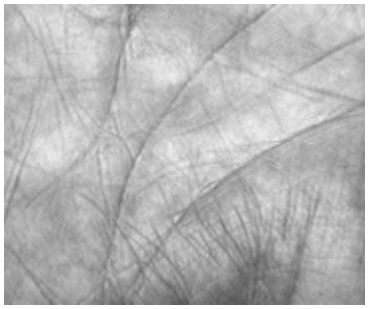

Figure 4.20 Central Area of a Sample Palm Image Obtained After Preprocessing

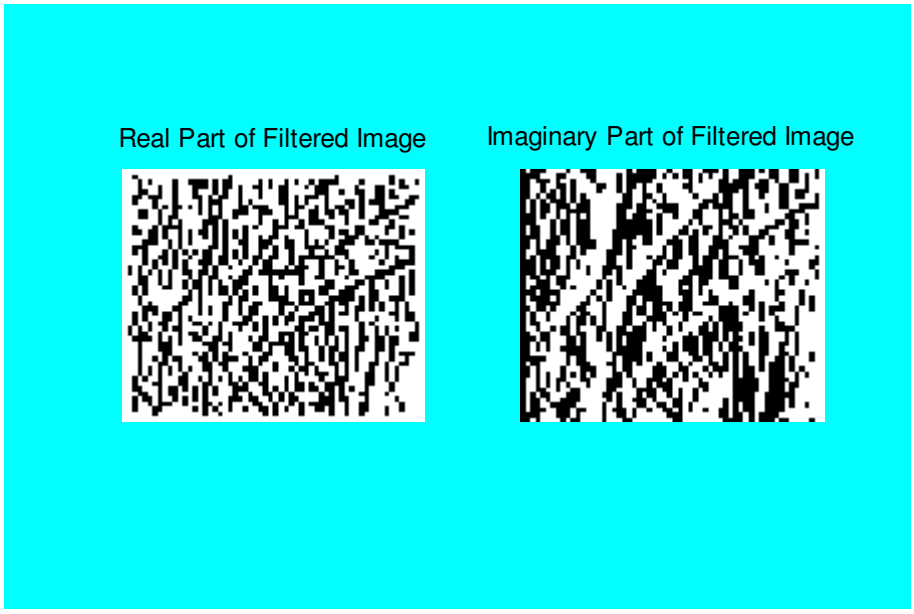

Figure 4.21 Real and Imaginary Parts of the Feature Vectors for the Image in Figure 4.20

In order to give a better understanding of encoding results, real and imaginary parts of the feature vectors are drawn as binary images in Figure 4.21. Many identification tests have been performed in order to see the identification accuracy when the imaginary component, b_{Imag}, the real component, b_{Real}, the phase component, $\arctan(b_{Imag} / b_{Real})$, and both the real and the imaginary component, b_{Real} and b_{Imag}, are used as feature vectors for different threshold values. Identification accuracy results obtained for different feature vectors and for different threshold values are shown in Figure 4.22. During these tests, we realized the fact that the real part does not carry any valuable information. This is because Gabor filter parameters are selected in the way that necessary information for recognizing palms is stored in the imaginary part of the feature vector. Therefore; there is no need in processing and storing the real part, which results in faster algorithm and smaller template size. Thus, the feature vector is reduced to the imaginary part only. Furthermore, the threshold value is selected as -0.2 based on the identification accuracy results obtained in these tests. Moreover, the selection of the threshold value as -0.2 is justified by ROC curves plotted for different threshold values when the imaginary part is employed as the feature vector. This is depicted in Figure 4.23. Finally, it is worth noting that principal lines in Figure 4.20 can also be seen in Figure 4.21, proving the correctness of the selected Gabor filter parameters in terms of accuracy.

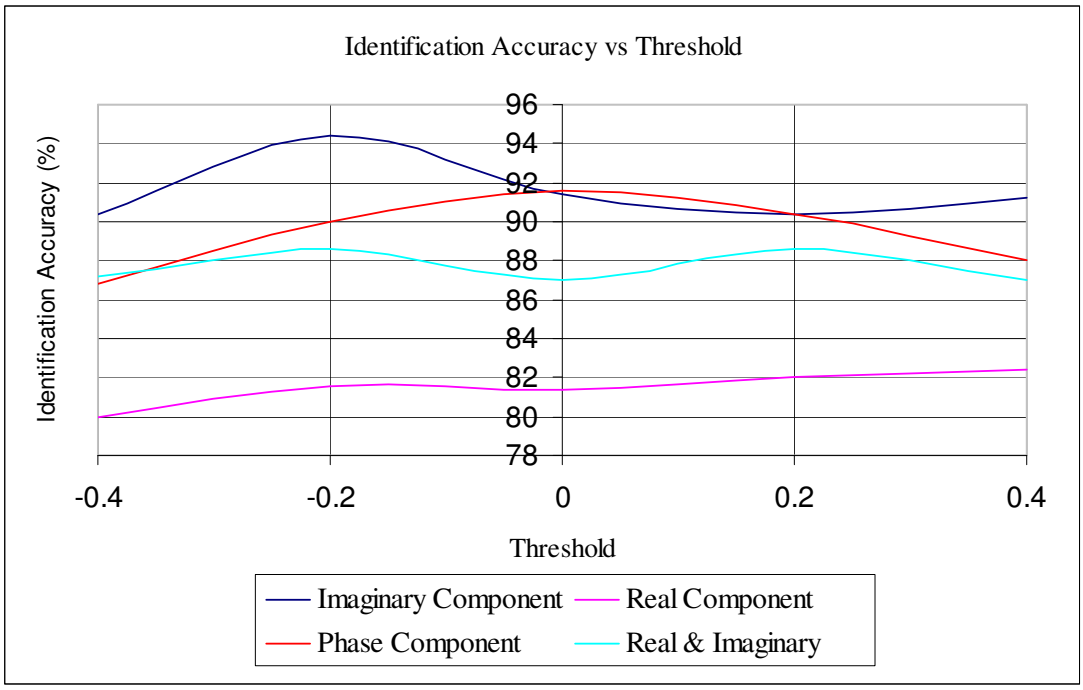

Figure 4.22 Identification Accuracy for Different Feature Vectors

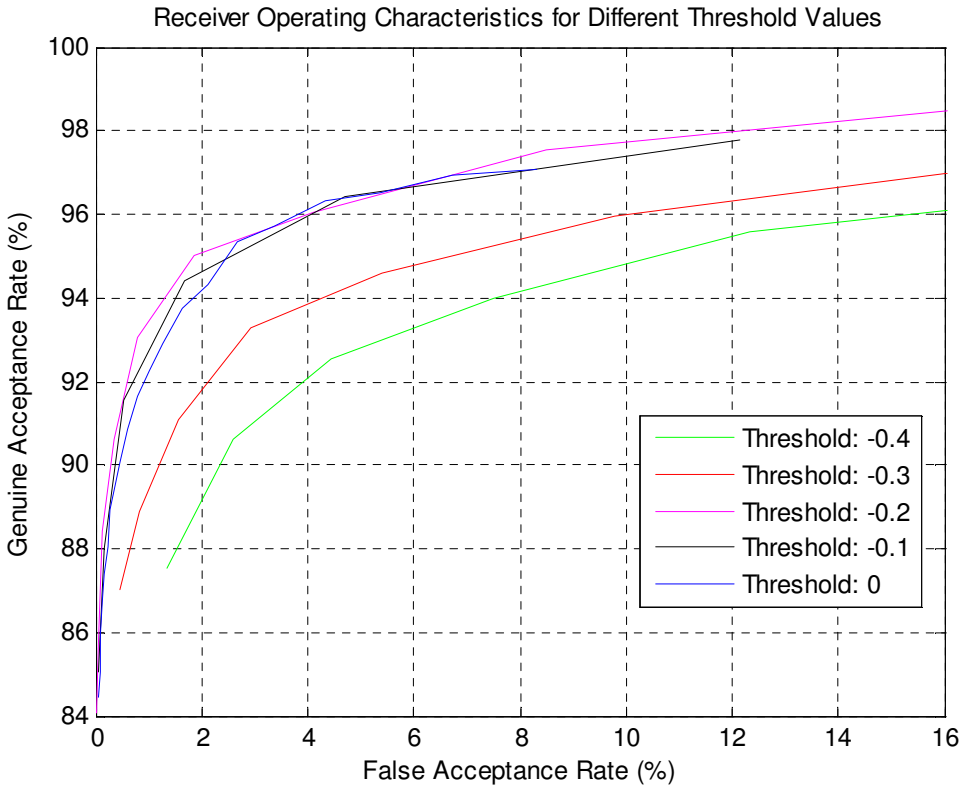

Figure 4.23 ROC Curve for Different Threshold Values

4.3 Distance Matching and Decision Policy

In the previous block, relevant features of the central palm area are extracted and coded, hence; the mathematical representation of the palm is obtained. In this block, the mathematical representations of many palm images are compared to each other and a decision is made based on the decision policy.

The first step in this block is to define a matching algorithm in order to measure the degree of similarity between two feature vectors. Due to the fact that, all features in a feature vector is of the same importance, there is no need in normalizing the feature vectors. In addition, 1-bit encoding performed in the previous block easies the selection of the matching algorithm. This is because, many different distance matching algorithms such as sum of absolute difference and sum of square distance reduce to the same distance matching algorithm which can be mathematically expressed as follows:

$$D = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} P(i,j) \oplus Q(i,j)}{2MN} \tag{4.12}$$

where P and Q are two feature vectors to be compared, $\oplus$ represent the bitwise exclusive-or operator, and M, N are dimensions of feature vectors.

Remembering the fact that bitwise exclusive-or operator generates 1 when two operands are different and 0 otherwise; D should be 0 for perfect matching; that is for identical feature vectors. On the other hand, when feature vectors are bitwise complements of each other, D reaches its maximum value, ½. Thus D is a rational number between 0 and ½, and the smaller D is, the more similar the feature vectors are.

However, some modifications are required on the distance matching algorithm stated in Equation (4.12). This is because; the algorithm in Equation (4.12) is based on pixel by pixel comparison, hence; it is very sensitive to vertical and horizontal

translations and rotations in any direction. These translations and rotations result from both the misplacement of palms during palm acquisition and the imperfect preprocessing. In order to take vertical and horizontal translations and possible rotations into account, the distance matching algorithm can be modified as follows:

$$D = \min_{s\leq|S|,t\leq|T|} \frac{\sum_{i=(\max(1,1+s)+S)}^{\min(N,N+s)-S} \sum_{j=(\max(1,1+t)+T)}^{\min(N,N+t)-T} P(i+s,j+t) \oplus Q(i,j)}{2H_s H_t} \tag{4.13}$$

where S = 7 and T = 7 determine the maximum allowable vertical and horizontal translation, and H_s and H_t in Equation (4.13) can be expressed as follows:

$$\begin{aligned} H_s &= \min(N, N+s) - \max(1,1+s) - 2S \\ H_t &= \min(N, N+t) - \max(1,1+t) - 2T \end{aligned} \tag{4.14}$$

The matching algorithm given in Equation (4.13) increases the resistance of the developed algorithm against rotation and translation and therefore; it results in a more robust algorithm. It is here worth noting that the proposed algorithm is still sensitive to big amount of rotations and translations.

After the matching algorithm has been selected, the next step is to determine the decision policy. As it is usual, decisions are made by comparing the result of the distance matching algorithm, D, to similarity threshold which is empirically determined. If D is smaller than the similarity threshold, two palm images are matched, otherwise they are not matched.

The palmprint recognition algorithm has been detailed in this chapter. The results of the proposed algorithm will be given in the next chapter. The next chapter will also discuss the developed palmprint acquisition system which offers some improvements, especially in preprocessing block.

CHAPTER V

RESULTS

This chapter details the results obtained on The Hong Kong Polytechnic University Palmprint Database from different side of views including accuracy, speed and template size. Moreover, it describes small modifications made on the developed algorithm after integration of the scanner in order to adapt the algorithm to palm images acquired via the scanner. The organization of this chapter is as follows: Section 5.1 details results on The Hong Kong Polytechnic University Palmprint Database. Section 5.2 describes the small modifications on the algorithm after the scanner is integrated and mentions results obtained in the new palmprint database formed in METU.

5.1 Results on The Hong Kong Polytechnic University Palmprint Database

5.1.1 Accuracy

5.1.1.1 Verification Accuracy

In order to obtain the verification accuracy of the algorithm, every palm image in the database is compared with each other. Since there are 600 palm images in the database, there are 179700 total comparisons. Because there are 6 palm images for each palm, in 1500 of these total comparisons palm images belonging to same palms are compared. In the remaining 178200 comparisons, palm images of different palms are compared. Ideally, a matching should only occur if two palm images belong to same palm and a non-matching should only occur if two palm images belong to different palms. However, because it is nearly impossible to design a biometric system with an accuracy of 100%, the proposed system results in incorrect matching,

in which two palm images of different palms are matched, and incorrect non-matching, in which two palm images belonging to same palm are not matched. Number of incorrect matchings is used in order to calculate FMR, Similarly, number of incorrect non-matchings is used to calculate FNMR.

Figure 5.1 displays the histogram of the distance between palm images belonging to same palm. Similarly, Figure 5.2 shows the histogram of the distance between palm images of different palms. Figure 5.3 shows the probability density function of genuine and imposter matching scores. Finally, the ROC curve of the proposed system has been plotted in Figure 5.4. As it may be seen in Figure 5.4, an Equal Error Rate of 3.82 % has been obtained.

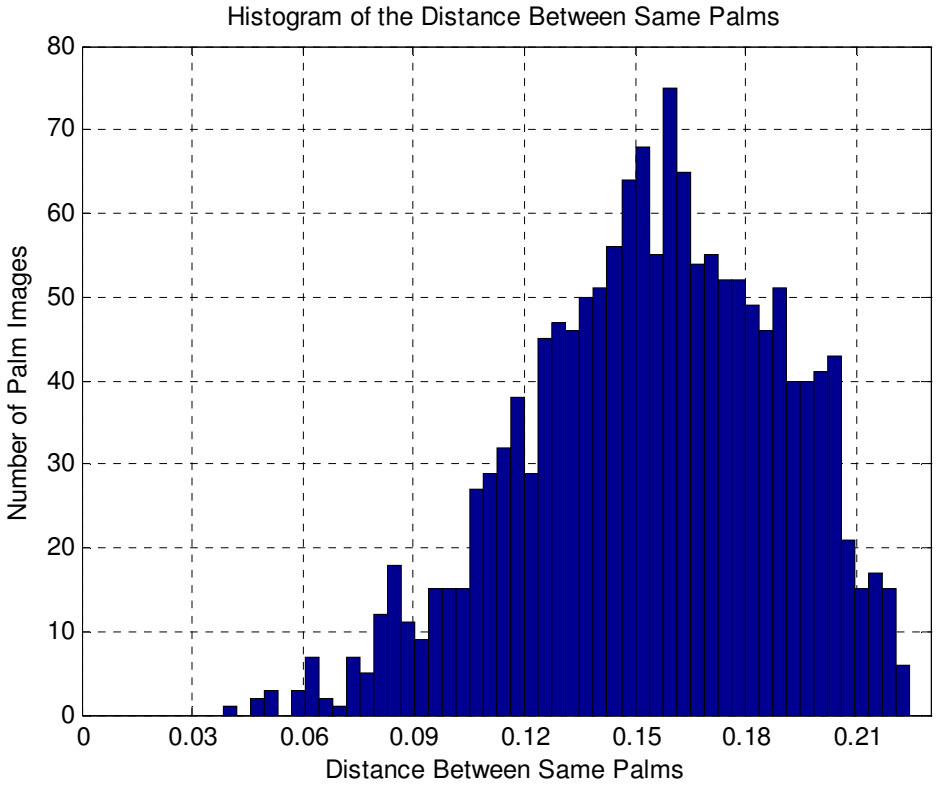

Figure 5.1 Histogram of the Distance Between Palm Images Belonging to Same Palm

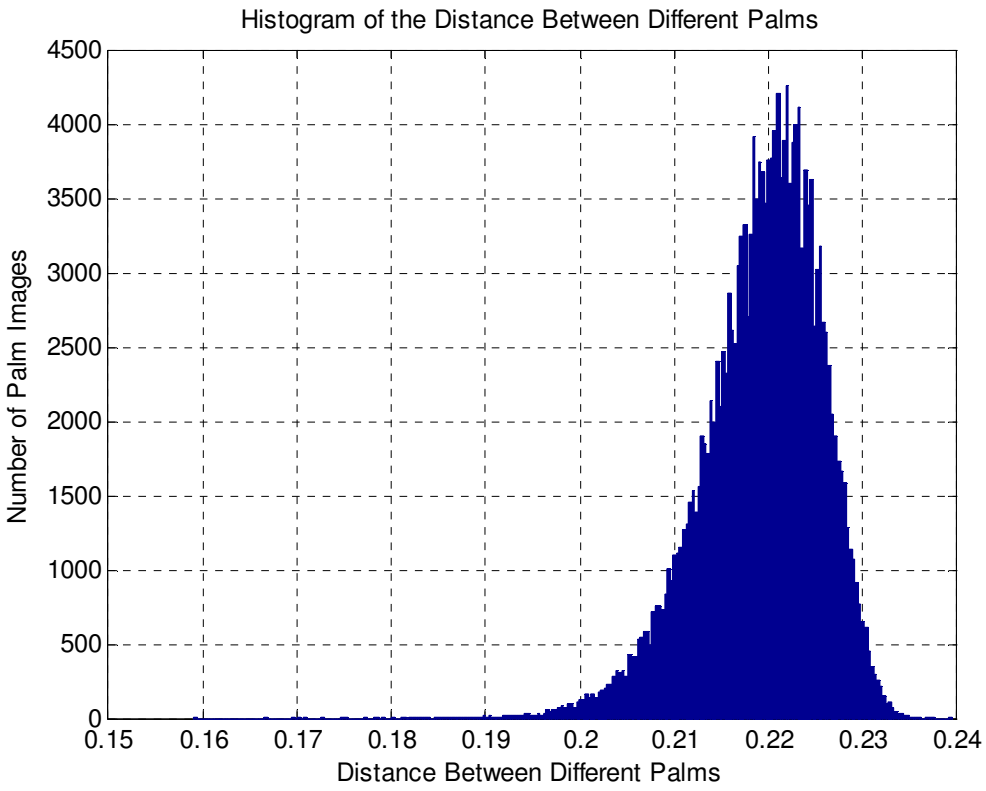

Figure 5.2 Histogram of the Distance Between Palm Images Belonging to Different Palms

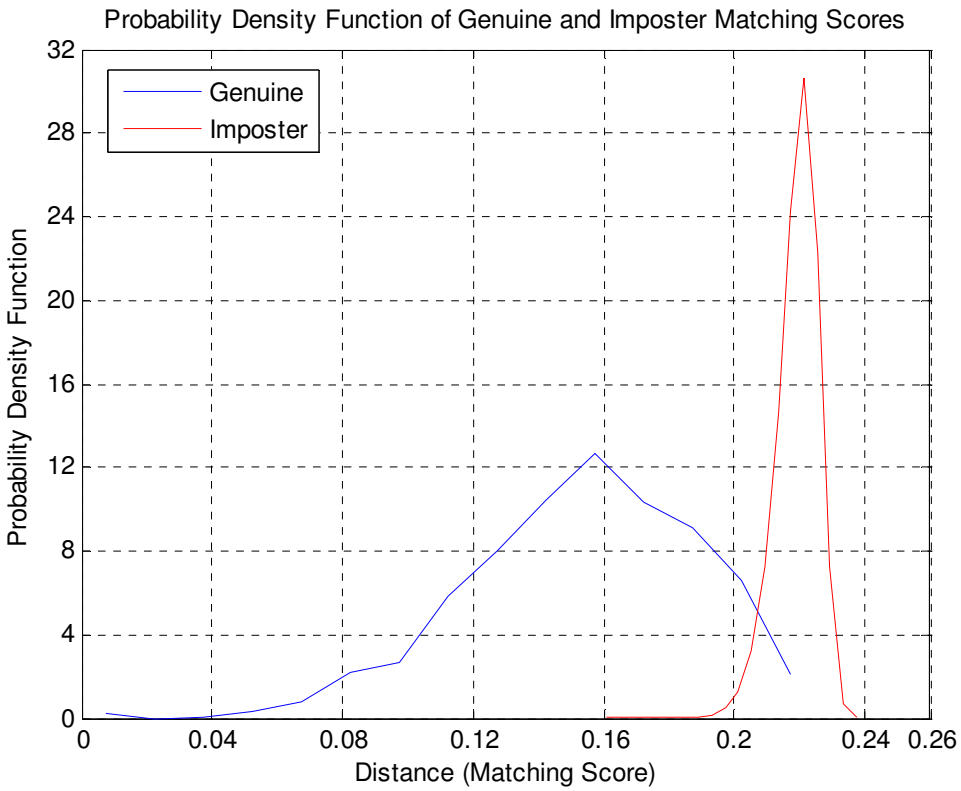

Figure 5.3 Probability Density Functions of Genuine and Imposter Matching Scores

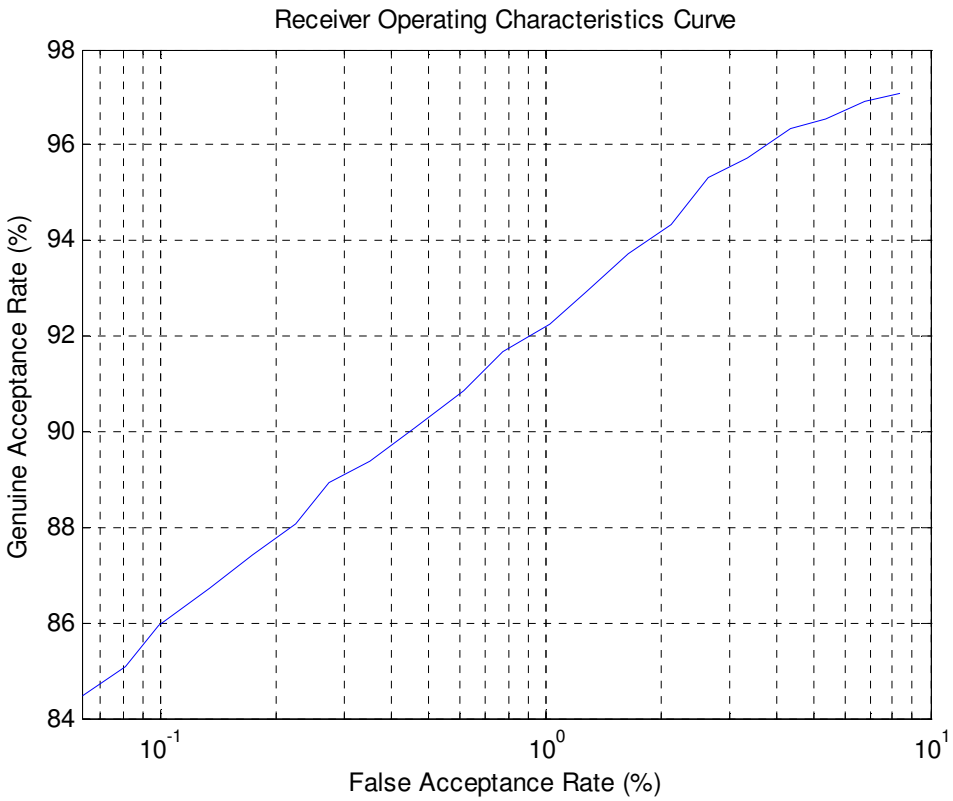

Figure 5.4 ROC Curve of the Proposed Algorithm

5.1.1.2 Identification Accuracy

In identification, a palm image has been compared to N palm images. For our particular case, one palm image for each palm; namely PolyU_xx_1.bmp where xx is the unique palm identifier (ranges from 00 to 99); is reserved as the template to be stored in the database. Remaining palm images; namely PolyU_xx_N.bmp where xx is again the unique palm identifier and N ranges from 2 to 6; are used as test images to be identified. Therefore; there are 1 template and 5 test images for each palm. Noting that xx ranges from 00 to 99; there are totally 100 templates and 500 test images. Each test image is compared to all templates in the database, and the test image is matched with the most similar template, that is the template generating the smallest matching score.

Figure 5.5 displays the histogram of the smallest distance, the distance between the test images and the most similar templates, for correct matches. Figure 5.6 shows the histogram of the second smallest distance, the distance between the test images and the second most similar templates. It is here worth noting that the difference between the smallest distance and the second smallest distance gives an idea about the reliability of the identification; that is the bigger the difference is, the more reliable the identification is. Let the reliability of identification ratio, RI, be defined as the ratio of this difference to the smallest distance, as in Equation (5.1). The histogram of the reliability of identification ratio is depicted in Figure 5.7.

$$\mathrm{RI} = \frac{\left(\text{Second Smallest Distance - Smallest Distance}\right)}{\text{Smallest Distance}} \tag{5.1}$$

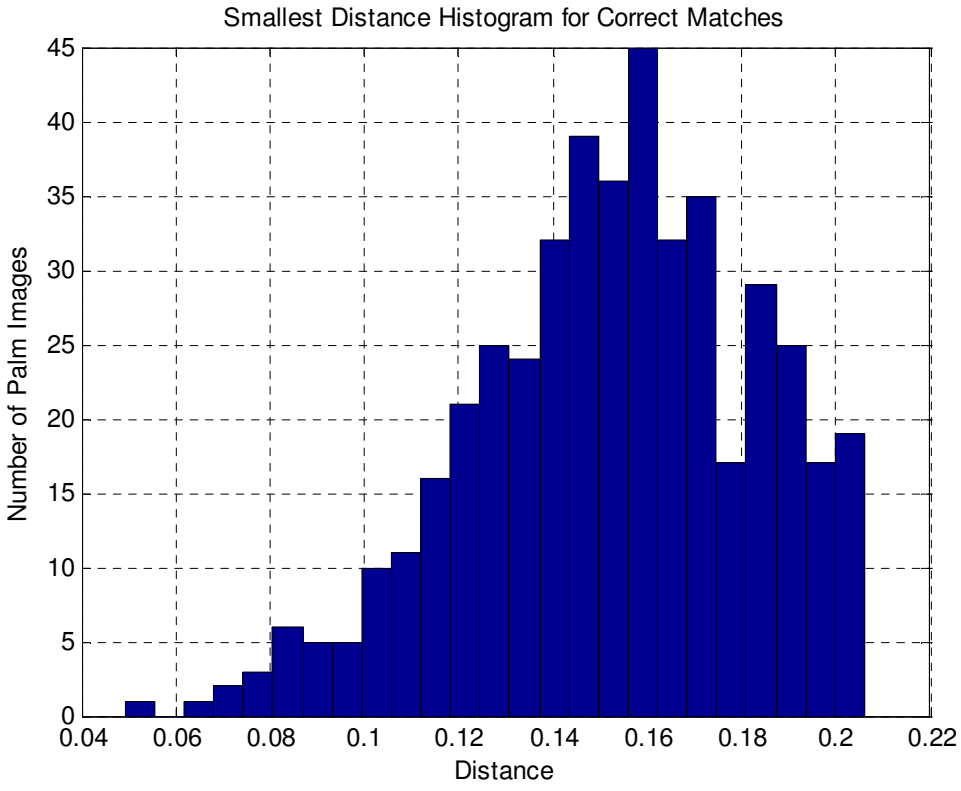

Figure 5.5 Smallest Distance Histogram for Correct Matches

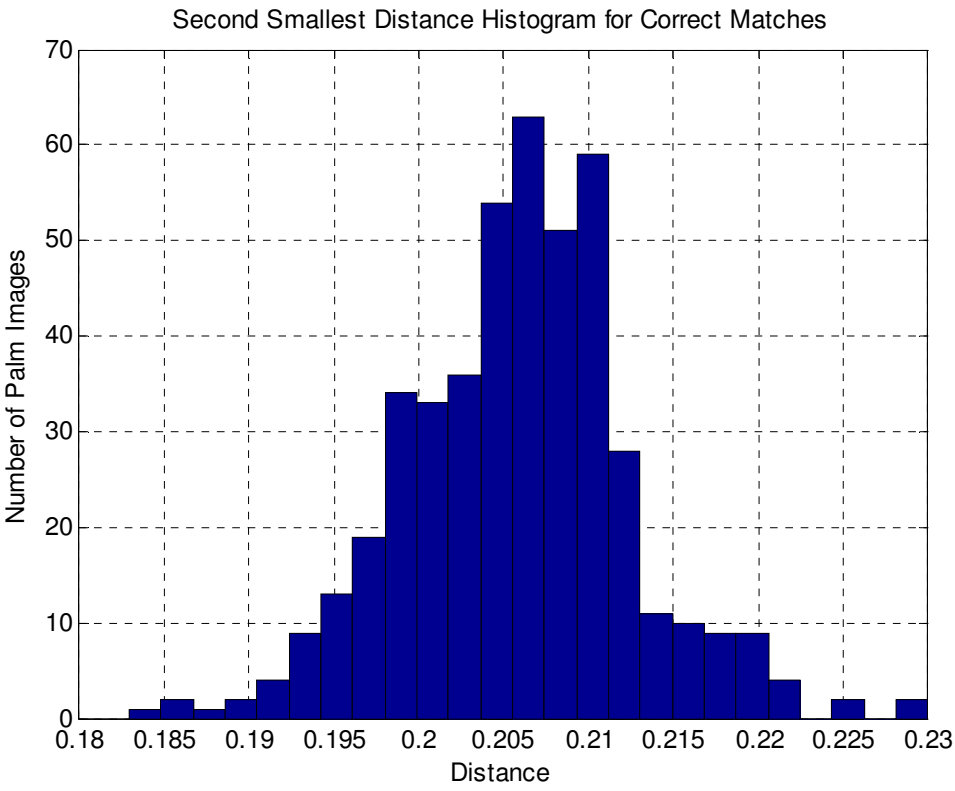

Figure 5.6 Second Smallest Distance Histogram for Correct Matches

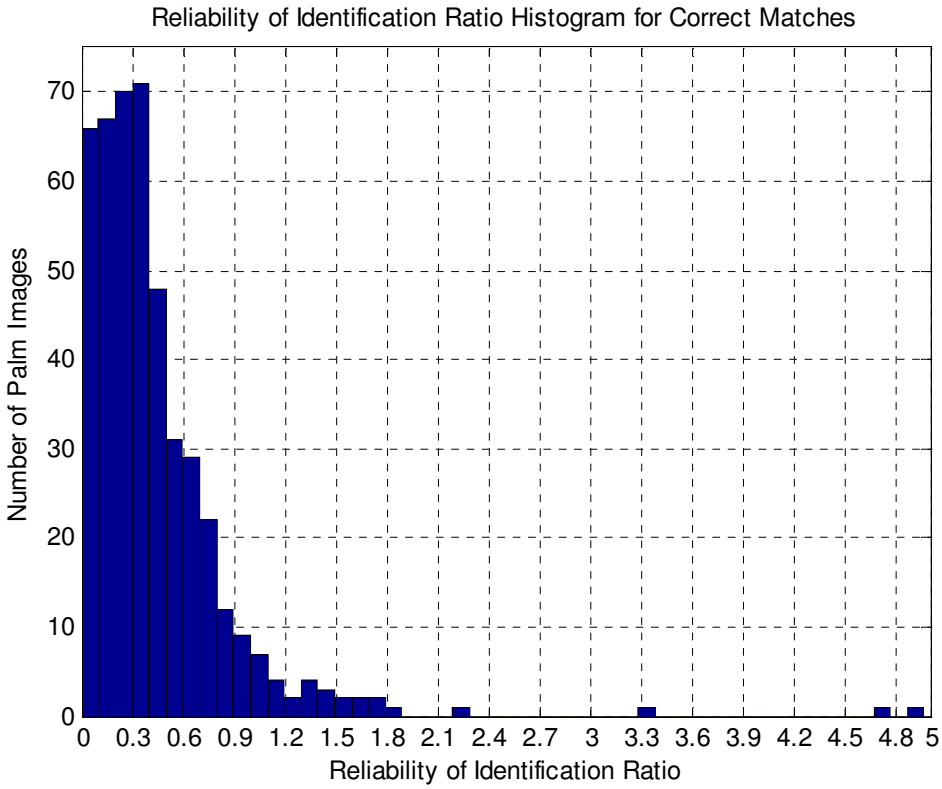

Figure 5.7 Reliability of Identification Ratio Histogram for Correct Matches

In biometric systems, there are usually three or more templates stored in the database for every individual and the biometric data to be identified is compared to all these templates. Then distances between the biometric data to be identified and all templates of an individual are used to produce a single matching distance to be used in decision policy by either taking the average or the minimum of these distances. Nonetheless; the proposed algorithm has reached to an acceptable correct identification accuracy of % 94.4, even with one template. It is obvious that, a higher correct identification level can be obtained if the number of templates stored in the database is increased. Nevertheless; the increase in the number of templates significantly decreases the number of test images and this may result in insufficient number of test images in order to judge the identification capability of the proposed system.

5.1.2 Template Size

Size of templates stored in the database is an important issue in the evaluation of a biometric system. Template size is extremely important for the biometric applications in which biometric data of many individuals need to be stored. In Chapter IV, it is stated that the feature vector in the proposed algorithm contains 3000 elements, each of 1 bit. Hence, the size of templates stored in the database is 375 bytes, which is quite small considering the fact that 1 million templates can be stored in a memory of size about 360 MB.

5.1.3 Speed

The proposed algorithm is implemented using MATLAB® 7.0 on a PC with a 1.6 GHz Intel Pentium M processor and a 512 MB RAM. The execution time for the preprocessing is about 0.75 second. The next block, namely feature extraction and coding block, executes in 0.3 second. Finally, the execution time for the distance matching and decision block is about 20 milliseconds. Since total verification time does not depend on the number of templates in the database, it can be found by summing up execution times of preprocessing block, feature extraction and coding block and distance matching block. Therefore; the total verification time is calculated to be around 1 second, which is acceptable for an online palmprint recognition algorithm. Total identification time, on the other hand, depends on the number of templates, since test images to be identified are compared to all templates in the database. Figure 5.8 shows the change of the total identification time with respect to the number of templates in the database. It is seen that the total identification time increases linearly with the number of templates as expected and it becomes nearly 3.1 seconds when the number of templates is 100. It is here worth noting that the algorithm has been implemented in MATLAB®, an interpreted language which is known to be a number of times slower than C/C++. Therefore, the total identification time can be significantly reduced by implementing the same algorithm in C/C++. It can also be seen in Chapter IV that the algorithm does not use built-in functions in MATLAB® Image Processing Toolbox® except Canny edge detector in order to

provide platform independence, that is, in order to ease the implementation of the same algorithm in other programming languages. In brief, the total identification time for 100 templates can be said to be still acceptable and the speed of the algorithm can be further increased by selecting a compiled language such as C/C++ to implement the algorithm.

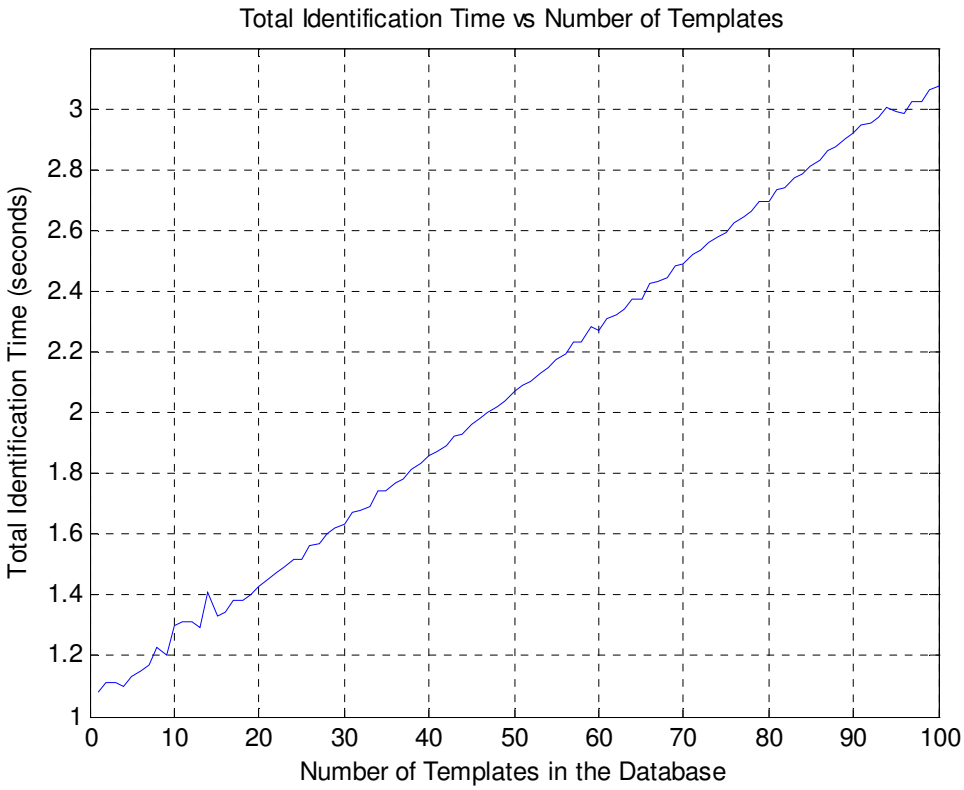

Figure 5.8 Total Identification Time versus Number of Templates in the Database

5.2 Palmprint Database Formed in METU

Performance of the developed algorithm on The Hong Kong Polytechnic University Palmprint Database is quite satisfactory. Nevertheless, success of the algorithm on this database may not be enough to claim that the developed algorithm is a generic palmprint recognition algorithm. In order to evaluate the performance of the algorithm on another palmprint database, a small database consisting of 28 different palms has been set up. A palmprint acquisition system employing a scanner has been

integrated to the developed algorithm. The scanner, HP Scanjet 3800, is controlled from MATLAB® environment by the trial version of Ciansoft TwainControlX, which is an ActiveX control that enables applications to acquire images from TWAIN compliant devices such as scanners and digital cameras. [28] The scanner is adjusted to scan images of 100 dpi resolution to test the algorithm with low resolution images. Figure 5.9 shows a sample palm image acquired via the scanner.

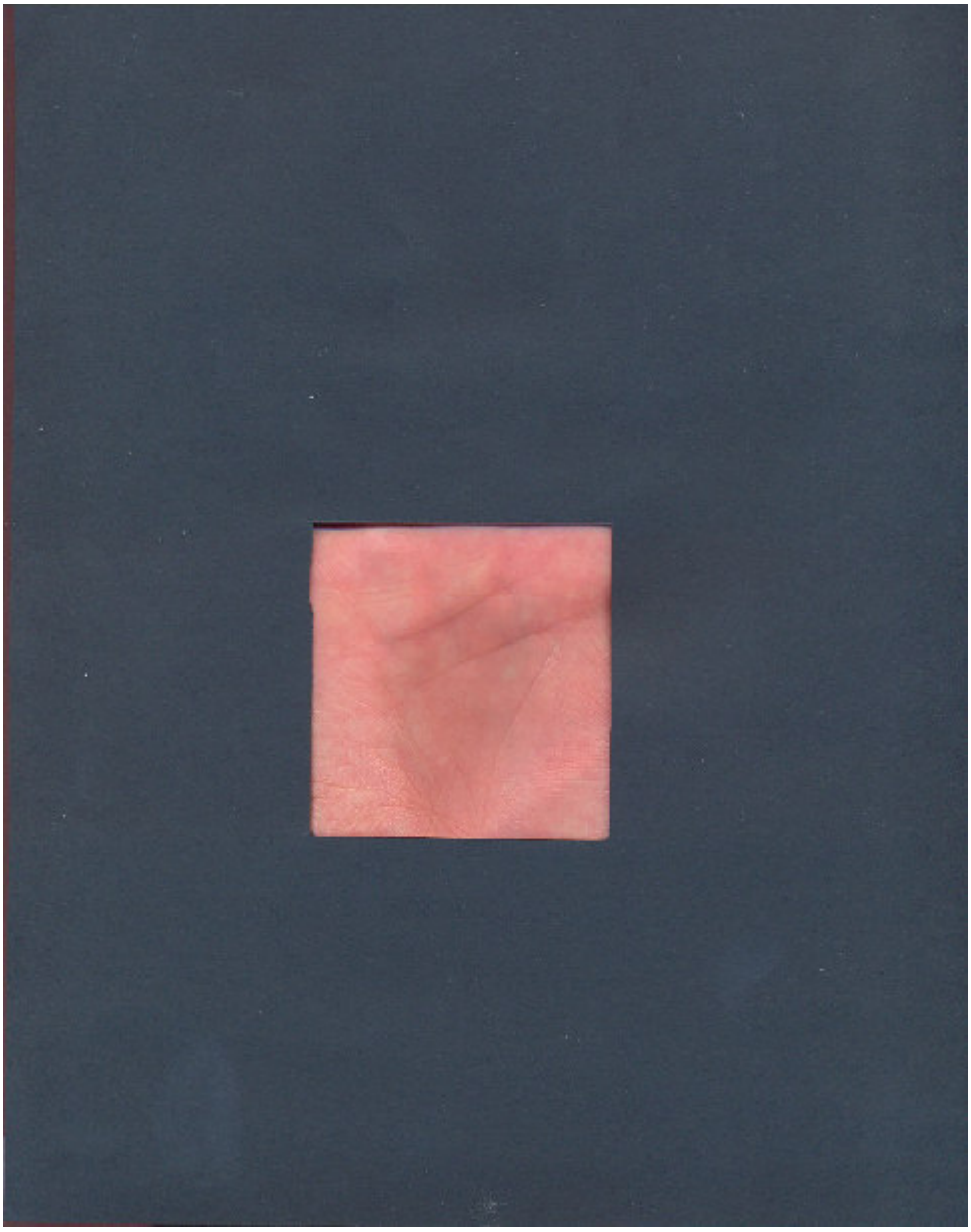

Figure 5.9 Sample Palm Image Acquired via the Scanner

5.2.1 Modifications in the Developed Algorithm

The preprocessing block has been modified in order to adopt changes in acquired palm images. In fact, the preprocessing block has been significantly simplified in order to reduce the execution time of this block. First, the central area of the image in Figure 5.9 is cropped and the resulting image is converted to grayscale image according to the NTSC standard:

$$I = 0.2989 * rgb\_img(:,:,1) + 0.5870 * rgb\_img(:,:,2) + 0.1140 * rgb\_img(:,:,3); \quad (5.2)$$

where I is the grayscale image and rgb_img(:,:,1), rgb_img(:,:,2) and rgb_img(:,:,3) represent the red, blue and green components of the RGB image, respectively. The resulting grayscale image of the image in Figure 5.9 is shown in Figure 5.10.

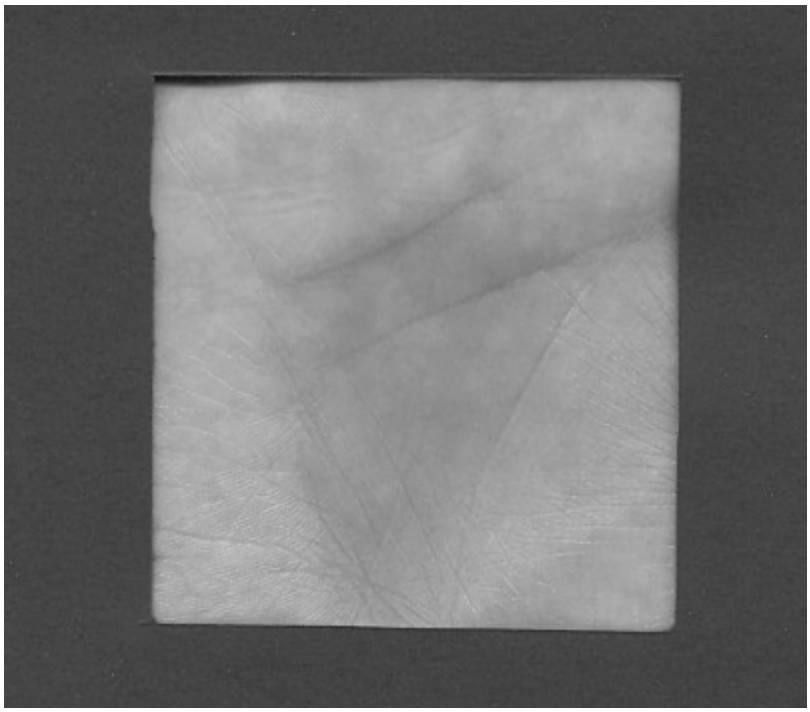

Figure 5.10 Cropped Grayscale Image

In order to separate relevant palm area from background, pixel values of the grayscale image in Figure 5.10 are compared to a threshold value of 150 and the desired palm area is obtained as shown in Figure 5.11.

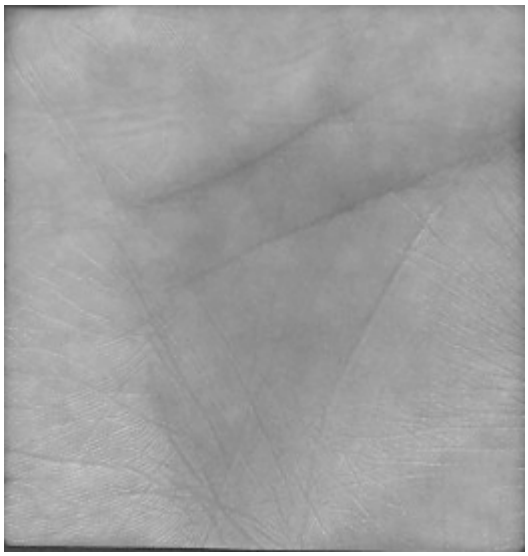

Figure 5.11 Desired Palm Area

The execution time for the preprocessing block is reduced to 50 milliseconds after simplifications. This results in significant decreases both in the total verification time and in the total identification time.

Although this new preprocessing block is different from the preprocessing block detailed in Section 4.1, the heart of the algorithm, that is the feature extraction and coding block and the distance matching block, is kept same except a slight modification in the feature extraction and coding block. The slight modification is that the convolution result of the Gabor filter and the central palm area is divided into sub-blocks of 4x4 instead of 3x3 and the mean value of the corresponding 16 pixels

is calculated to be compared with the same threshold value of -0.2. This very small difference in the feature extraction and the coding block has no effect on the accuracy of the algorithm and it just aims to decrease the size of the templates to be stored in the database.

5.2.2 Results on New Database

As it is the case in the identification tests performed on The Hong Kong Polytechnic University Palmprint Database, only one palm image is stored in the new database for each individual. Many identification tests have been performed on this new database and Figure 5.12 depicts the matching score graph obtained in one of these identification tests. The identification accuracy obtained on The Hong Kong Polytechnic University Palmprint Database is improved to around %98 on this new database. Since the algorithm is same, the improvement in the identification accuracy can be said to be caused by better preprocessing block. Since individuals who provide palm images can see the area on which they place their palms, they have the opportunity to place their palms in a more correct way. It can also be added that errors caused by imperfect preprocessing on The Hong Kong Polytechnic University Palmprint Database are reduced on the new database by the new simple preprocessing block. Finally, it is here worth noting that the number of different palms in the new database is less than the number of different palms in The Hong Kong Polytechnic University Palmprint Database and this most probably has a positive effect on the increase in the identification accuracy.

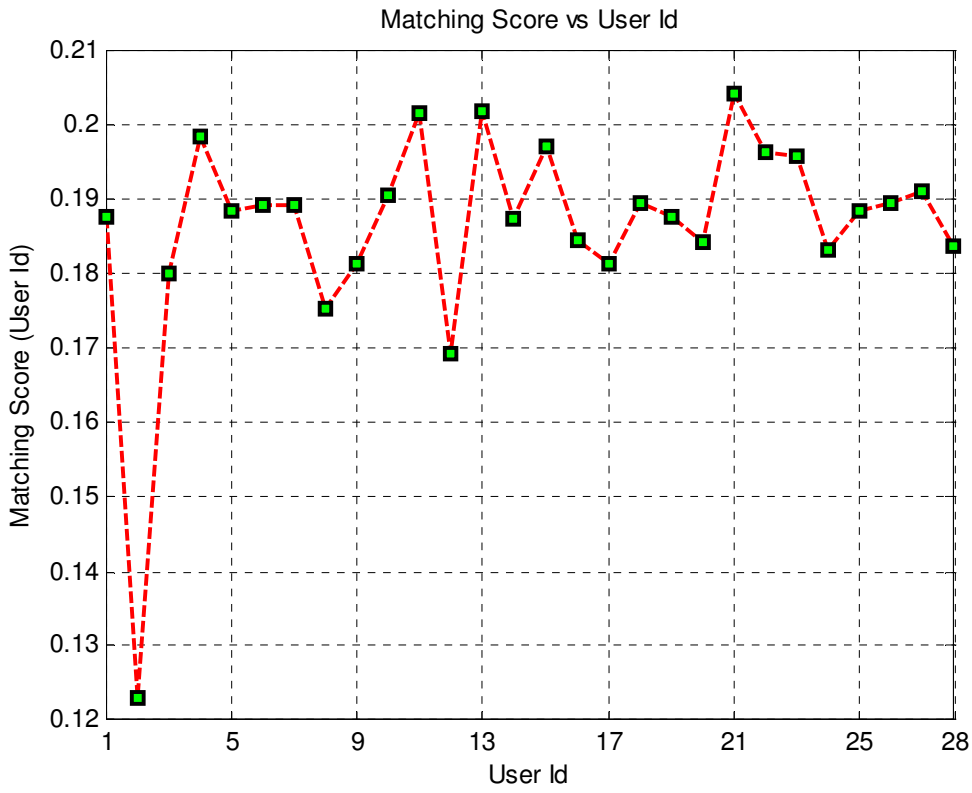

Figure 5.12 Sample Matching Score Graph

5.3 Summary of Results

As it is seen, EER of the proposed algorithm obtained on The Hong Kong Polytechnic University, % 3.82, is comparable to that of existing palmprint recognition algorithms in the literature, which ranges from % 0.3 to % 14. Furthermore, the identification accuracy obtained on the same database even with one template, % 94.4, is also comparable to that of existing palmprint recognition algorithms, which ranges from %90 to % 99. It is here worth adding that most of the existing palmprint recognition algorithms use three or more templates during identification tests. In addition, the identification accuracy of the proposed algorithm obtained on the palmprint database formed in METU is even higher, around % 98. Moreover, speed of the algorithm in both verification and identification is acceptable for an online palmprint recognition system and it can be further increased by selecting a compiled language in order to implement the same algorithm. Because built-in functions in MATLAB® Image Processing Toolbox® are almost not utilized,

implementing the same algorithm in any other programming language is relatively easy. In this sense, the proposed algorithm provides platform independence. Furthermore, the proposed algorithm is very efficient in coding in terms of template size. It is able to encode larger palm area in smaller size than existing palmprint recognition algorithms. Finally, it is worth adding that the proposed algorithm can work quite accurately with low resolution palm images.

CHAPTER VI

CONCLUSION

Reliable authentication is very critical in today's world. Among three authentication methods; namely knowledge based authentication, password based authentication and biometric based authentication; biometric based authentication is known to be the most reliable method. Being the most reliable authentication method together with the strong need in reliable authentication resulted in an exponential increase in biometric revenues in recent years. Researchers realizing this increase have been trying to develop more accurate algorithms for existing biometrics such as fingerprint, face, iris and voice. Moreover, they are searching for new biometrics that can provide high accuracy.

Palmprint is a relatively new field of biometrics. Nevertheless, palmprint-scan technology has many advantages over other biometric technologies. It provides high accuracy and enables utilization of cheap acquisition devices. In addition, it provides simple user-system interaction, therefore; zero or very small FTER can be obtained. Furthermore, palmprint-scan technology has a very high user acceptance, that is, there is a little objection, if any, to using a palmprint-based biometric authentication. These noticeable advantages of palmprint-scan technology resulted in a considerable increase in palmprint recognition algorithms, especially in last three years. Recognition rates between %90 and % 99 have been obtained in these studies. It is here worth noting that there is a problem related to common palmprint database usage and this is mainly due to the fact that palmprint recognition is a relatively new research area. However, The Hong Kong Polytechnic University Palmprint Database is the most commonly used palmprint database.

In this study, a palmprint recognition algorithm based on 2-D Gabor filters has been implemented in MATLAB® environment. This algorithm consists of three main

blocks, namely preprocessing block, feature extraction and coding block and distance matching and decision policy block. In preprocessing block, central area of the palm has been extracted. In the subsequent block, feature extraction and coding block, central palm area is filtered with a 2-D complex Gabor filter. Then the imaginary part of the filtered image is encoded as it is detailed in Equation (4.11), and the feature vector has been obtained. In the last block, distance matching and decision policy block, feature vectors are compared to each other using the distance matching algorithm given in Equation (4.13) and the final decision is made based on the distance between feature vectors and the threshold which is determined empirically. During the development of the algorithm, built-in functions in MATLAB® Image Processing Toolbox® have not been utilized except Canny edge detector in order to provide platform independence. Therefore, the same algorithm can be implemented in any other programming language easily.

Developed algorithm is first tested on The Hong Kong Polytechnic University Palmprint Database. During verification tests, EER of % 3.82 has been obtained. Total execution time in verification is around 1 second, fast enough to be used in real-time applications. In addition, identification accuracy of 94.4 % has been obtained even with one template stored in the database for each palm. Were the number of templates stored in the database for each palm higher, higher identification accuracy could be obtained. Nevertheless, increasing number of templates decreases number of test images further and this may result in meaningless identification accuracy due to small number of test images to be identified. Total identification time is around 3 seconds for 100 templates and it increases linearly with the number of templates. Although it is still acceptable for on-line identification systems, total identification time can be further decreased by implementing the same algorithm in a compiled language such as C or C++. It is expected to be easy to implement the same algorithm in any other language due to reasons mentioned above. Finally, size of templates stored is 375 bytes, quite small in today's technology.

In order to test the performance of the developed algorithm on another database, a scanner is integrated to the developed algorithm in MATLAB® environment. With

this palmprint recognition system, a new palmprint database consisting of 28 different palms is formed in METU. Preprocessing block is modified in order to adopt changes in acquired palm images. However, feature extraction and coding block and distance matching and decision policy block have remained same except a very small modification in the feature extraction and coding block. Identification accuracy of 98 % has been obtained on this new database. The improvement in identification accuracy is most probably caused by better preprocessing block.

The developed algorithm can work accurately with low resolution images. In addition, the algorithm is robust against changes in illumination because of usage of Gabor filter. The algorithm is also robust to small amount of vertical and horizontal translations and small amount of rotations mainly because of the utilized distance matching algorithm which takes small variations into account. However, since the algorithm is based on pixel by pixel comparison, the algorithm is sensitive to big amount of translations in both direction and big amount of rotations. Nevertheless, big amount of variations both in translation and rotation can be avoided by informing enrollees.

As a future work, the proposed algorithm can be implemented on an embedded system, that is, on a microprocessor or on a field programmable gate array (FPGA) platform. Furthermore, 2-D Gabor filter utilized in the proposed algorithm can be replaced with a Gabor filter bank, in which each Gabor filter is oriented in different direction. Moreover, number of palm images in the palmprint database formed in METU can be increased and this palmprint database may be published, making it freely available for academic purposes, in order to provide researchers developing palmprint recognition algorithms the opportunity to compare their algorithms to existing palmprint recognition algorithms. Finally, the palmprint recognition system can be improved by replacing the scanner with a smaller and faster palm acquisition device. After this replacement, the palmprint recognition system may be a commercial product which can be used in industry.

REFERENCES

[1] John D. Woodward, Jr., Nicholas M. Orlans, Peter T. Higgins, "Biometrics", McGraw-Hill, 2003.

[2] Ruud M. Bolle, Jonathan H. Connell, Sharath Pankanti, Nalini K. Ratha, Andrew W. Senior, "Guide to Biometrics", Springer, 2004.

[3] Samir Nanavati, Michael Thieme, Raj Nanavati, "Biometrics", Wiley, 2002.

[4] "Biometrics Glossary", retrieved 10 January 2007, from http://www.biometricscatalog.org/biometrics/GlossaryDec2005.pdf

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

[6] David Zhang, Wai-Kin Kong, Jane You, Michael Wong, "Online Palmprint Identification", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, No. 9, pp 1041-1050, September 2003.

[7] M. Wong, D. Zhang, W.-K. Kong and G. Lu, "Real-time Palmprint Acquisition System Design", IEE Proc.-Vis. Image Signal Process., Vol. 152, No. 5, October 2005.

[8] Fang Li, Maylor K.H. Leung, Xiaozhou You, "Palmprint Identification Using Hausdorff Distance", 2004 IEEE International Workshop on Biomedical Circuits & Systems, 2004.

[9] Fang Li, Maylor K.H. Leung, Xiaozhou You, “Palmprint Matching Using Line Features”, ICACT 2006, 20-22 February 2006.

[10] N. Duta, A.K. Jain, “Matching of Palmprints”, Pattern Recognition Letters I, Vol. 23, pp 477-485, 2002.

[11] J. You, W. X. Li and D. Zhang, “Hierarchical Palmprint Identification via Mutiple Feature Extraction”, Pattern Recognition Vol. 35, No. 4, pp 847-859, 2002.

[12] D. Zhang, W. Shu, “Two Novel Characteristics in Palmprint Verification: Datum Point Invariance and Line Feature Matching”, Pattern Recognition, vol 33, no. 4, pp 691-702, 1999.

[13] Li Shang, De-Shuang Huang, Ji-Xiang Du, Chun-Hou Zheng, “Palmprint Recognition using FastICA algorithm and radial basis probabilistic neural network”, 2005.

[14] Xiang-Qian Wu, Kuan-Quan Wang, David Zhang, “Palmprint Recognition Using Valley Features”, Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2005.

[15] A.K. Jain, S. Prabhakar, L. Hong, S. Pankanti, “Filterbank-Based Fingerprint Matching”, IEEE Trans. Image Process., Vol. 9, No. 5, pp 846–859, 2000.

[16] L. Hong, Y. Wan, A. Jain, “Fingerprint Image Enhancement Algorithm and Performance Evaluation”, IEEE Trans. Pattern Anal. Mach. Intell., Vol. 20, No. 8, pp 777–789, 1998.

[17] C.J. Lee, S.D. Wang, “Fingerprint Feature Extraction Using Gabor Filters”, Electron. Lett., Vol. 35, No. 4, pp 288–290, 1999.

[18] Xiangqian Wu, Kuanquan Wang, Fengmiao Zhang, David Zhang, “Fusion of Phase and Orientation Information for Palmprint Authentication”, 2005.

[19] Ajay Kumar, Helen C. Shen, “Palmprint Identification using PalmCodes”, Proceedings of the Third International Conference on Image and Graphics, 2004.

[20] Wai King Kong, David Zhang, Wenxin Li, “Palmprint Feature Extraction Using 2-D Gabor Filters”, Pattern Recognition, Vol. 36, pp 2339- 2347, 2003.

[21] M.J. Lyons, J. Budynek, S. Akamatsu, “Automatic Classification of Single Facial Images”, IEEE Trans. Pattern Anal. Mach. Intell., Vol. 21, No. 12, pp 1357–1362, 1999.

[22] B. Duc, S. Fischer, J. Bigun, “Face Authentication with Gabor Information on Deformable Graphs”, IEEE Trans. Image Process., Vol. 8, No. 4, pp 504–516, 1999.

[23] Y. Adini, Y. Moses, S. Ullman, “Face Recognition: The Problem of Compensation for Changes in Illumination Direction”, IEEE Trans. Pattern Anal. Mach. Intell., Vol. 19, No. 7, pp 721–732, 1997.

[24] “2-D Gabor Function – interactive visualization”, retrieved 17 November 2006, from http://www.cs.rug.nl/~imaging/simplecell.html

[25] “Gabor Filter”, retrieved 13 November 2006, from http://en.wikipedia.org/wiki/Gabor_filter

[26] “Detecting bars and lines in images”, retrieved 19 November 2006, from http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/AV0405/SIKLOSSY/bars.html

[27] “Two-Dimensional Gabor Filters”, retrieved 22 November 2006, from http://zerodeux.net/perso/DEA/irisscan/PAMIpaper/node5.html

[28] “TwainControlX”, retrieved 20 September 2006, from http://www.ciansoft.com/twaincontrolx/default.asp