

SECURING BIOMETRIC DATA

BY

AHMAD HUSSEIN

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

Computer Science

January 2010

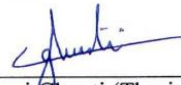
KING FAHD UNIVERSITY OF PEROLEUM & MINERALS

DHAHRAN 31261, SAUDI ARABIA

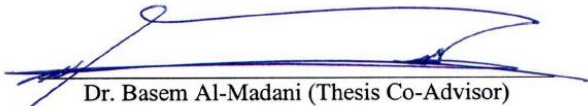
DEANSHIP OF GRADUATE STUDIES

This thesis, written by **AHMAD MAHMOUD KHALIL HUSSEIN** under the direction of his thesis advisor and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN COMPUTER SCIENCE**.

Thesis Committee



Dr. Lahouari Ghouti (Thesis Advisor)



Dr. Basem Al-Madani (Thesis Co-Advisor)



Dr. Kanaan A. Faisal
Department Chairman



Dr. Salam A. Zummo
Dean of Graduate Studies



Dr. Khaled Saleh (Member)



Dr. Wasfi Al-Khatib (Member)



Dr. Samir H. Abdul-Jauwad (Member)

7/2/10

Date



To my great parents

Acknowledgment

Acknowledgment is due to King Fahd University of Petroleum & Minerals for supporting this research.

First of all, I am very grateful to my parents whom after Allah were the source of my guidance. I will never forget their unconditional support and continuous prayer. Second, I wish to express my appreciation to Dr. Lahouari Ghouti who served as my thesis advisor and in the same time as a brother. Also, I am very thankful to Dr. Basem Al-Madani, my thesis co-advisor, for his kind cooperation. The valuable feedback and comments of the thesis committee members, Dr. Khaled Salah, Dr. Wasfi Al-Khatib and Dr. Samir Abdul-Jauwad, have significantly improved the quality of this thesis. Their time and efforts are highly appreciated. Thanks to my brothers and sister for their support and encouragement. I am also very thankful to all individuals who have helped me during my M.S. study at King Fahd University of Petroleum and Minerals.

Table of Contents

Table of Contents	iv
List of Figures	vi
List of Tables	x
THESIS ABSTRACT	xi
ملخص الرسالة.....	xiii
CHAPTER 1: Introduction	1
1.1 Introduction	1
1.2 Problem Statement	10
1.3 Thesis Objectives	12
1.4 Thesis Contributions	14
1.5 Thesis Organization	15
CHAPTER 2: Literature Review	16
2.1 Introduction	16
2.2 Fuzzy-vault Techniques	19
2.3 Biometric Certification Techniques	25
2.4 Watermarking Techniques	32
2.5 Comparative Study of the Existing Techniques	49
CHAPTER 3: Mathematical Preliminaries	52
3.1 Iris Representation	53
3.2 Wavelet Transform	63

3.3	Wavelet Modeling	72
3.4	Edge Process (EP) Model	76
CHAPTER 4: Proposed Techniques		82
4.1	Fuzzy-vault for Iris Templates	82
4.2	Fingerprint Data Watermarking	92
4.3	Variations in the Proposed Algorithms	102
CHAPTER 5: Experimental Results.....		104
5.1	Introduction.....	104
5.2	Experiments and Data Setup	106
5.3	Performance Results	115
5.4	Conclusions and Summary.....	135
CHAPTER 6: Conclusions		136
6.1	Summary	136
6.2	Future Work	137
REFERENCES.....		139
VITAE		149

List of Figures

Figure 1: Biometric attributes.	4
Figure 2: Biometric system functionalities [1].	7
Figure 3: Typical performance measures of biometric systems. (a) FAR vs. FRR. (b) FRR vs. FAR (c) GAR vs. FAR [1].	10
Figure 4: Points for attacking biometric system [3].	18
Figure 5: Fuzzy-vault example [6].	21
Figure 6: Fingerprint minutiae points.	21
Figure 7: Iris and other eye components [10].	23
Figure 8: Vault encoding and decoding.	24
Figure 9: EyeCerts System [12].	27
Figure 10: FaceCerts system [13].	29
Figure 11: Eigenface example.	31
Figure 12: A sample bank note of 20 US \$ [18].	34
Figure 13: Typical digital watermarking system. a) Watermark encoding stage. b) Watermark decoding stage [19].	37
Figure 14: Compressed domain fingerprint watermarking a) Input fingerprint b) Fingerprint with data embedded [25].	43
Figure 15: Fragile fingerprint watermarking a) Watermark b) Watermarked fingerprint image [26].	44
Figure 16: First technique for biometric watermarking [33].	48
Figure 17: Second technique for biometric watermarking [33].	49
Figure 18: a) iris image b) edge map c) horizontal edge map d) vertical edge map [10].	55

Figure 19: a) Iris image. b) Segmented iris image.	56
Figure 20: The remapping formula [10].	59
Figure 21: Iris normalization [10].	60
Figure 22: a) Real part characterized by a cosine modulated by a Gaussian b) Imaginary part characterized by a sine modulated by a Gaussian [10].	62
Figure 23: Feature encoding using 1D Gabor filters [10].	63
Figure 24: Windowed Fourier Transform (WFT) window [35].	65
Figure 25: Daubechies Wavelet in time-frequency plane [35].	66
Figure 26: Different Wavelet families [35].	67
Figure 27: Wavelet Transform Process.	70
Figure 28: Example of a sample image.	71
Figure 29: The Wavelet transform of the image in Figure 28.	71
Figure 30: Example of two different iris codes.	83
Figure 31: Vault encoding phase.	84
Figure 32: Example of encoded vault. The blue points are the key points and the red points are random chaff points.	87
Figure 33: Vault decoding phase.	90
Figure 34: Fuzzy-vault for multiple instances of one iris image.	91
Figure 35: Fuzzy-vault for right and left irises.	91
Figure 36: Example of bad quality fingerprint image (left side) and its enhancement using STFT analysis (right side).	93
Figure 37: Singular point detection using Poincare index.	94
Figure 38: Fingerprint image after removing the extra parts using SUSAN algorithm.	95
Figure 39: A) Image after resizing. B) Image after applying DWT.	96
Figure 40: Edges determined using edge process model.	97
Figure 41: Proposed watermarking system.	98

Figure 42: a) Original image b) Watermarked image.....	101
Figure 43: Minutiae points a) Original image b) Enhanced image c) Watermarked image using EP.	102
Figure 44: a) sample of normalized iris b) edges map after using the EP model.....	103
Figure 45: Image filtering effects on perceived image quality.	108
Figure 46: Set of fingerprint images used.....	112
Figure 47: Filtering effects of window-based filters.	113
Figure 48: Filtering effects of non-window filters.....	113
Figure 49: Blurring effects on sample iris image.....	115
Figure 50: Motion effects on iris images.....	117
Figure 51: Sharpening effects on iris images.	119
Figure 52: Performance of fuzzy-vault-based iris template protection under various filtering attacks.....	120
Figure 53: Effects of windowed filters on a sample fingerprint image. (a) STD filtering. (b) Median filtering. (c) Wiener filtering.	123
Figure 54: Watermark decoding performance in the presence of STD filtering attack.....	124
Figure 55: Watermark decoding performance in the presence of median filtering attack..	124
Figure 56: Watermark decoding performance in the presence of Wiener filtering attack. .	125
Figure 57: JPEG compression effect. (a) Original image. (b) JPEG compressed image using quality factor of 100. (c) JPEG compressed image using quality factor of 20.....	126
Figure 58: Watermark decoding performance in the presence of JPEG compression attack.	127
Figure 59: The effect of the third category filters on a watermarked fingerprint image. (a) Effect of blurring filtering attack. (b) Effect of rotation filtering attack. (c) Effect of motion filtering attack.	128
Figure 60: Watermark decoding performance in the presence of blurring filtering attack.	129
Figure 61: Watermark decoding performance in the presence of rotation filtering attack.	130
Figure 62: Watermark decoding performance in the presence of motion filtering attack. .	130

Figure 63: Implementation diagram of EyeCerts® algorithm.	133
Figure 64: Implementation diagram of FaceCerts® algorithm.....	134

List of Tables

Table 1: Comparison for biometric template security techniques. 50

THESIS ABSTRACT

Name: Ahmad Mahmoud Khalil Hussein

Title: Securing Biometric Data

Major Field: Computer Science

Date of Degree: January 2010

The emergence of biometric-based person identification in various applications, ranging from border security to simple access control, has opened the floor to several issues. Authentication and certification of biometric content stand at the forefront of these issues. Although praised for their robustness and reliability, (multi)biometric systems, like their ID and password-based counterparts, are hindered by their vulnerabilities to malicious attacks and intentional manipulations. Such alterations would nullify their usability for legal and juridical purposes.

In this thesis, we address several issues related to the certification of biometric data. First of all, we define the levels of biometric attacks and protection measures to mitigate their effects. The robustness of fuzzy

vault (FV)-based protection of biometric templates is investigated in depth. Iris templates are considered in this work. Also, a new security scheme to protect fingerprint data is proposed. Unlike existing watermarking-based protection schemes, the proposed one discreetly embeds protection payload in salient features of fingerprint images. These salient features are estimated using statistical modeling of fingerprint edges using the statistical edge process (EP). Moreover, the proposed watermarking scheme is characterized by increased robustness against attacks due to the "selective" embedding process adopted.

ملخص الرسالة

الإسم: أحمد محمود خليل حسين

عنوان الرسالة: حماية بيانات أنظمة التعرف الحيوية

التخصص: علم الحاسوب و المعلومات

تاريخ التخرج: 20 \ 1 \ 2010 م

إزدادت أهمية التعرف على الشخصية في ك ثير من تطبيقات الحياة، التعرف على الشخصية باستخدام الصفات الحيوية أصبح أكثر أهمية من سوابقه ابتداءً من التحكم في إعطاء الأولوية إلى غيرها من تطبيقاتها المعقدة، التعرف على الشخصية بواسطة الأنظمة الحيوية كان من أهم الطرق في تحديد الهوية، و بالرغم من قوة هذه الأنظمة و سه ولة الإعتماد عليها فإن لها بعض نقاط الضعف التي يسهل إختراقها كسابقتها من أنظمة التعرف التقليدية، إذ أن أي إختراق في هذه الأنظمة يمكن أن يلغي عملها و يجعلها غير صالحة للإستعمال في بعض الدوائر القانونية.

في هذه الرسالة، سوف نقوم بتحديد بعض النقاط المهمة المتعلقة بأنظمة التعرف الحيوية المعترفة، أولاً: سوف نحدد نقاط الضعف و بعض مقاييس الأمان لها، حيث أن قوة العقد القطنية قد قيمت بعمق بإستخدامها في حماية شيفرة قزحية العين، كما أن خوارزمية جديدة إقتُرحت لحماية بيانات بصمة الإصبع بإستخدام العلامات المائية، التعليم المائي المستخدم يعتمد على إخفاء البيانات في المناطق البارزة في صور بصمة الإصبع، المناطق البارزة حددت بواسطة نموذج إحصائي لمعالجة الحدود، كما أن قوة الخوارزمية المقترحة قد قيمت إتجاه بعض الإختراقات المختارة لفحص قوتها.

CHAPTER 1

INTRODUCTION

1.1 Introduction

Nowadays personal identity is becoming a very important task in governmental and personal procedures like border control, accessing sensitive data, bank transactions, etc. An important issue in personal identity is to identify a person on a property that cannot be stolen or shared such as biometric traits like the fingerprint and iris features. This identification process is known as biometric authentication. Traditional personal identification methods rely on the use of passwords or personal identity (ID) cards. However, these methods suffer from severe limitations. For instance, passwords could be forgotten or manipulated. ID cards can be spoofed or stolen. Moreover, the holder of these authentication documents can claim that he/she was not involved in any

specific malicious transaction being investigated. This means that all traditional authentication techniques cannot be utilized for negative recognition and non-repudiation. Negative recognition enables biometric systems to determine whether a certain individual is indeed enrolled in the system. Non-repudiation provides tangible evidence on the access to specific resources/services by an individual which he/she cannot deny at later stages.

There is a great need for reliable and authenticated identity management systems that can “reliably” accommodate a large number of individuals or users. Biometric systems can be used for verification and identification. Verification is the process of deciding whether the identity holder is or not the person having the acquired biometric traits and features. Identifying the identity of the person whose biometric traits are being acquired represents the identification process. The latter process uniquely depends on the acquired biometric traits without resorting to any extra knowledge about the person’s identity. In both processes, reliability enables improved recognition rates.

1.1.1 Biometric Systems

Biometrics represents the science and technology of recognizing the identity of a person based on his/her own physical or behavioral attributes. Physical attributes include fingerprints, iris, palmprint, hand geometry, and deoxyribonucleic acid (DNA). Speech, keystrokes, gait, and signature are classified as behavioral attributes [1]. Figure 1 gives a summary of major physical or behavioral biometric traits commonly found in biometric systems.

Without loss of generality, biometric systems consist of five components: 1) biometric sensor(s); 2) feature extractor; 3) feature matcher; 4) template (feature) database; and 5) decision maker. It should be noted that some of these components may not be available in specific biometric systems. Some real-world systems include a quality assessment component which evaluates the quality of the acquired biometric raw data. This component is necessary to ensure the “fitness” of the acquired data for further processing to yield reliable decisions.

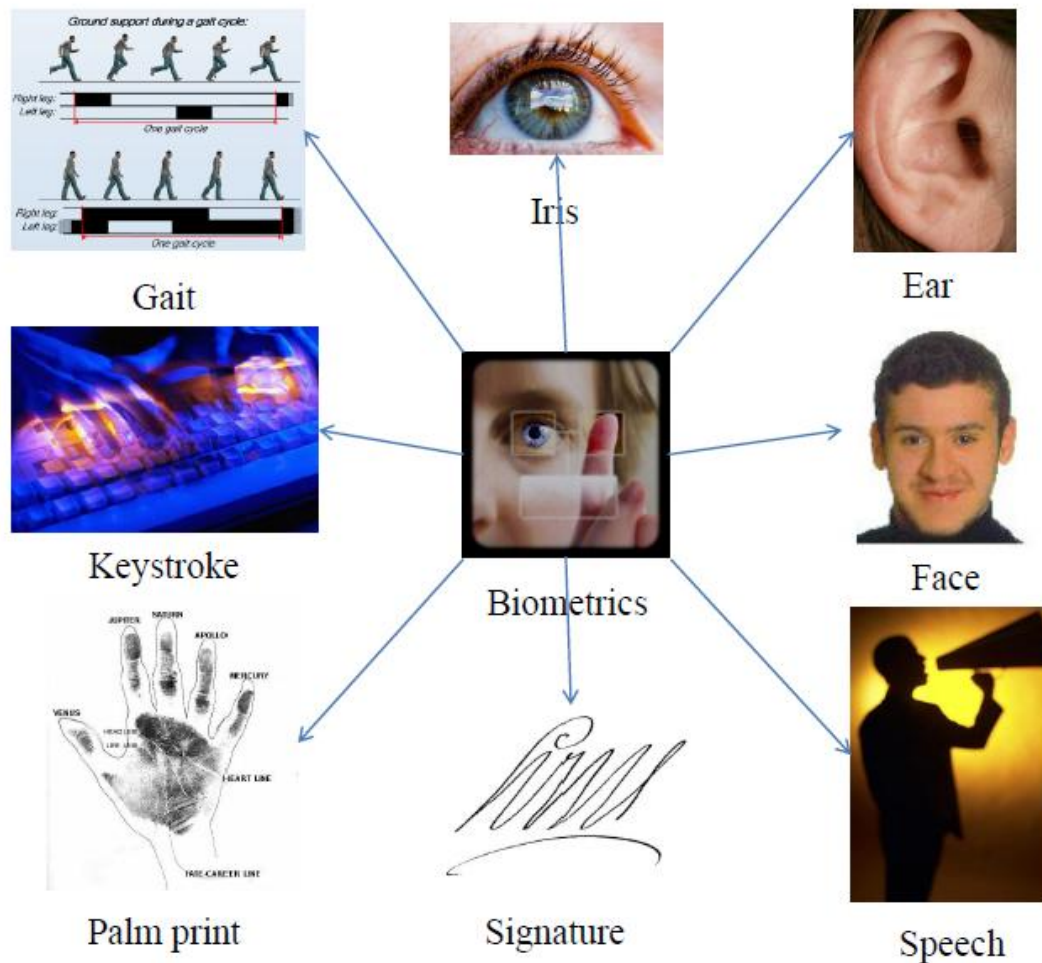


Figure 1: Biometric attributes.

Biometric sensors (readers or scanners) are used to acquire the raw biometric data. Fingerprint scanners and face/iris cameras are commonly called biometric sensors. The feature extractor module extracts the important features pertaining to the raw biometric data. Feature extraction (or reduction) enables “fault-tolerant” biometric systems operating at faster processing rates. Multi-dimensional transformations, pattern

recognition and data reduction techniques are usually found in such components.

The matcher module operates in the “feature-domain” and performs comparisons between the extracted feature and biometric templates stored in a (multi)-biometric database. Usually, template biometric databases contain stored biometric templates of enrolled persons. Finally, the comparison scores are evaluated by the decision maker module using specific metrics for matching and verification purposes.

Depending on their implementation, biometric systems can be either uni-modal or multi-modal depending on the number of “different” biometric traits being used. Multiple traits are considered for improving the system performance in terms of accuracy at the expense of increased computational payload.

1.1.2 Biometric System Functionalities

Biometric systems are design to operate in the following modes [1]:

- 1- Enrollment Mode: A common procedure usually taking place at an early stage. On need basis, a new person is registered (or enrolled)

into the system. Enrollment mode involves the acquisition (sensing), feature extraction, and storage processes.

2- Verification Mode: Usually, this mode takes place at the deployment stage. Using the biometric trait(s), a person identity is validated against his/her biometric template(s) stored in the database. To “biometrically” verify a person’s identity, sensing, feature extraction, matching, and decision making modules are utilized.

3- Identification Mode: In this mode, one-to-many comparisons are carried out to search through all templates of all users in the database for a possible match. Identification mode involves the acquisition (sensing), feature extraction, storage and decision making processes.

The possible functionalities of biometric systems are illustrated in Figure 2.

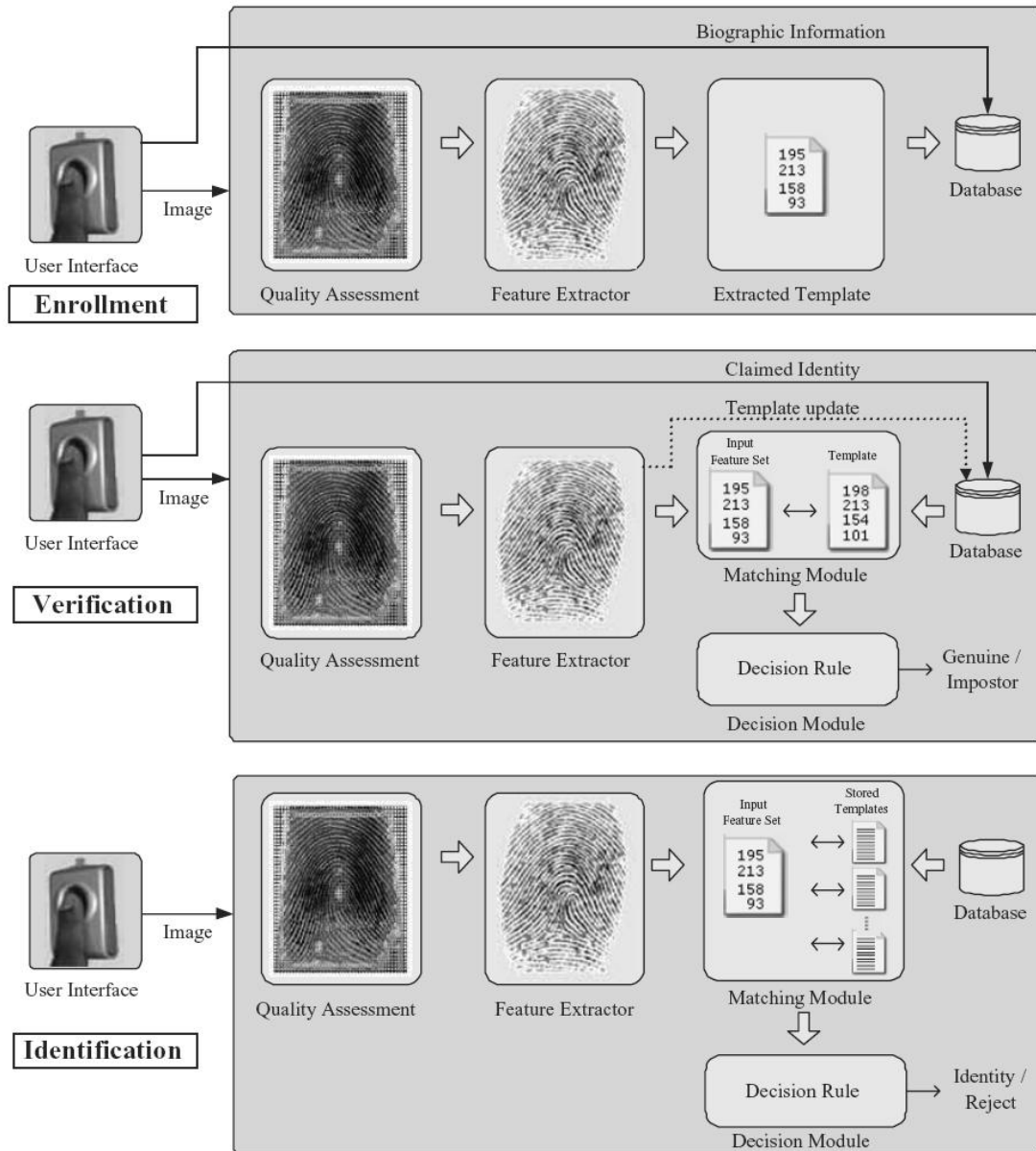


Figure 2: Biometric system functionalities [1].

1.1.3 Biometric System Performance

It is worth noting that multiple biometric traits of a single user taken at different times vary in several aspects. These variations in a single user's traits are commonly known as *intra-user variations*. For instance, in iris-based systems, a user may move his eyelids while capturing an image or the size of the iris may differ according to the surrounding light intensity and lighting conditions. Moreover, in fingerprint-based systems, a user may press or move his/her finger differently while his/her fingerprint is being captured. On the other hand, features extracted from biometric traits pertaining to different users can be quite similar. For example, some pairs of individuals can have nearly identical facial appearance due to genetic factors such as parenthood relations, identical twins, etc. This similarity between the biometric traits of different individuals is called *inter-user similarity*.

Due to these similarities and variations, biometric systems can make two types of errors; false accept (FA) and false reject (FR). If the intra-user variations are very large, the two traits from the same user will not be accepted which represents an FR error. Similarly, if two traits from two

different users are considered matching, then an FA error has occurred.

The latter situation often happens when the inter-user similarity is large.

Using the two types of errors mentioned above, the basic measures for the system performance are [1]:

- 1- False Accept Rate (FAR): The FAR represents the proportion of matches between two biometric traits from two different users that are incorrectly recognized as match.
- 2- False Reject Rate (FRR): The FRR represents the fraction of matches between two biometric traits from the same user that are recognized as pertaining to two different users.
- 3- Genuine Accept Rate (GAR): The GAR represents the proportion of matches between two biometric traits from the same user that are correctly recognized as match.

A typical performance measure using these metrics is shown in Figure

3.

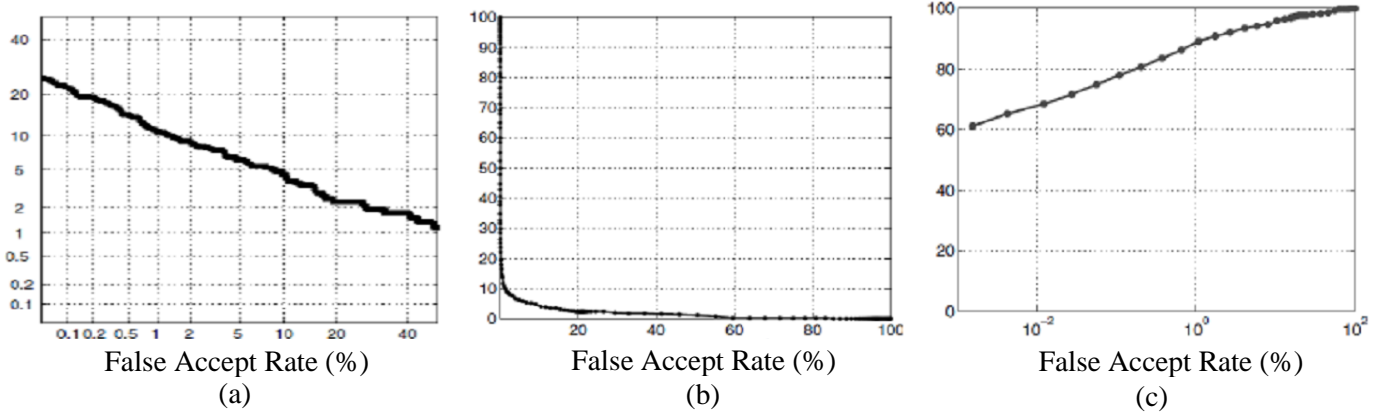


Figure 3: Typical performance measures of biometric systems. (a) FAR vs. FRR. (b) FRR vs. FAR (c) GAR vs. FAR [1].

1.2 Problem Statement

With the emergence of digital evidence and biometric data, person identification and forensic content have been constantly put under doubt for their vulnerability to digital forgery and seamless manipulations. Recently, safety and security of biometric data and templates have been the focus of many researchers due to their impact of the legal acceptance of “digital evidence” [2]. It is quite surprising to note that although biometric systems are proposed to increase the security of specific premises and applications, their own security is questioned at several levels. Moreover, due to their private nature, the identity of biometric content needs to be protected from illegal access and usage. At this stage,

a second level of security and protection is addressed. Also, unlike conventional passwords, which are usually protected through hard encryption, biometric data are usually presented in “clear” or “plain” format. Therefore, a potential attack on biometric content and templates would be more tempting due to the target easiness. For illustration, a person’s face can be easily captured with or without his consent. At various stages, a biometric authentication system is “openly” vulnerable to attacks. In response to these issues, this thesis proposes new protection and security techniques to increase the robustness of biometric data and templates against attacks and manipulations. Cryptography-based fuzzy vault scheme is proposed to secure iris templates. To assess the robustness of the proposed, a series of experiments are conducted to investigate the parameters and system thresholds necessary for such a system to “safely” operate under normal conditions. A watermarking-based protection scheme is propose to secure fingerprint images. It is worth noting that fingerprint-based biometric systems dominate the biometric market [1]. Because fingerprint ridges are crucial for the estimation of minutiae points [3], the proposed scheme discreetly embeds watermark payload in edges, the most salient features of a fingerprint image.

1.3 Thesis Objectives

As mentioned above, biometric data or templates need to be secured for several reasons. The thesis objectives encompass the security and certification of biometric data (and or templates). More specifically, the main objectives of this thesis are:

- 1- Investigate the security of the fuzzy vault (FV) scheme [4] through a detailed cryptanalysis study. The security of biometric templates will be considered in depth and system performance is analyzed in the presence of several malicious attacks and intentional tamper manipulations.
- 2- Propose a novel scheme to secure biometric data using a robust digital watermarking technique that involves salient biometric features such as fingerprint edges. The robustness of the proposed security scheme is thoroughly investigated in the presence of several malicious attacks and intentional tamper manipulations such as image filtering, compression and geometric processing.
- 3- Investigate the suitability of the security schemes outlined above depending on the application at hand.

- 4- Investigate the security of biometric data without resorting to the use of biometric databases. Like storing the biometric template on smart cards instead of using database as the case in EyeCerts® and FaceCerts® biometric systems.

Due to the diversity of the biometric templates, we will restrict our investigation to the security of biometric templates extracted from iris images for the fuzzy-vault scheme. In the case of securing biometric data using digital watermarking, only fingerprint images are considered due to their suitability to the proposed data embedding technique. In addition, the proposed technique will be benchmarked against existing techniques used to secure fingerprint images. Finally, for the last thesis objective, two well-established certification algorithms: EyeCerts® and FaceCerts® are considered. In both cases, there is no template database. However, the biometric data is stored on a separate module, namely a biometric card.

1.4 Thesis Contributions

The main contributions for this thesis are summarized as follows:

- 1- Demonstrate the robustness of iris-based fuzzy-vault scheme against typical template attacks where various image filtering techniques are carried out on iris test samples.
- 2- Propose a new security scheme for watermarking fingerprint data using Edge Process (EP) model.
- 3- Apply an in-depth cryptanalysis to the new proposed fingerprint watermarking system to demonstrate its improved performance. Many image alteration techniques have been applied through typical image filters.
- 4- Compare the applicability of the security schemes for securing biometric templates, biometric data, and certified biometric systems.
- 5- Demonstrate the functionality of two certified biometric systems and highlight their contribution to the protection of biometric systems against template database attacks.

1.5 Thesis Organization

The thesis is organized as follows: Chapter II provides a detailed review of existing techniques used to secure biometrics templates and data; certify biometric systems. A comparative study of the existing techniques is given as a summary at the end of this Chapter. Mathematical preliminaries for wavelet techniques, edge processing, and wavelet modeling are discussed in Chapter III. The main contributions and proposed techniques are detailed in Chapter IV. Chapter V discusses the experiments and computer simulations carried out in this thesis. A detailed summary of the performance results is given therein. Finally, this thesis concludes with Chapter VI where a summary of the main thesis contributions are outlined along with directions for possible future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In Chapter 1, it was indicated that authentication is becoming an increasingly important task in many applications and the consequences of an insecure authentication method can be catastrophic. In the previous Chapter, biometric systems and their importance in authentication process have been discussed. In this Chapter, the security of biometric systems and the ways for securing biometric systems have been discussed.

Ratha et al. [3] have identified eight points of attacking biometric systems as shown in Figure 4. Figure 4 shows that point 1 is an attacking point, an attacker can spoof a biometric trait like wearing a contact lenses

or silicon finger to get unauthorized access. While at points 2, 4, 6 and 8, an attacker can penetrate the communication channel to get data and replay it with another data or interrupt the channel. Feature extractor or matcher can be overridden with a Trojan horse to produce incorrect results as the case at points 3 and 5. At point 7, an attacker can modify the contents of the database. He can modify the biometric templates (feature points) or the raw biometric data (biometric images).

There are many techniques for securing these points. At point 1, sensing finger or eye conductivity or pulse can be a good approach for securing this point. Accordingly making all the components at points 3, 5 and 7 to reside in a secure location can help to protect them from unauthorized access. Furthermore, securing biometric templates can be used for point 7. At points 4, 6 and 8, these attacks can be prevented by using encrypted communication channels.

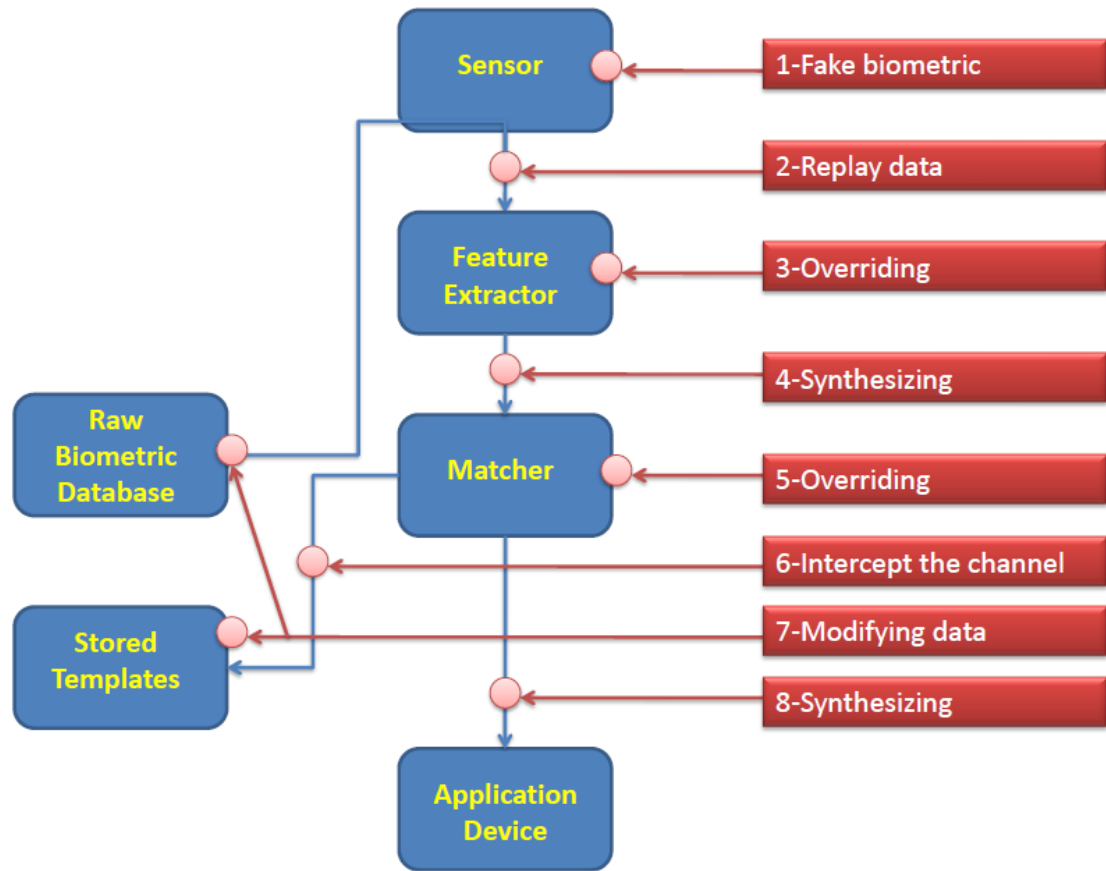


Figure 4: Points for attacking biometric system [3].

The work in this thesis is concentrated on point 7, how to secure biometric data and templates. Three methods for securing biometric data have been discussed as the following: fuzzy-vault for securing biometric templates, watermarking for securing biometric data, and the certification of biometric systems.

2.2 Fuzzy-vault Techniques

Template database can store biometric templates such as fingerprint minutiae points, facial features, or iris code. Fuzz-vault scheme can be used to secure biometric templates. Fuzzy-vault is a cryptographic technique. Encryption and fuzzy-vault are both members of cryptography.

Biometric encryption is a method to encrypt feature points of biometric data. There are famous algorithms that can be used for biometric encryption such as Rivest-Shamir-Adelman (RSA) algorithm, Advanced Encryption Standard (AES), etc. Jain et al. [5] revised methods of biometric encryption and their advantages and disadvantages. They stated that standard encryption techniques (like RSA and AES) are not useful for securing biometric templates. They indicated that matching cannot be done in encrypted domain because a small difference in decrypted data will result in a big difference in encrypted data. So for each authentication process data will be exposed. Biometric encryption has not been discussed deeply in this thesis.

Fuzzy-vault is a method of cryptography to secure any set of data using a polynomial function of a certain degree. This function called key. Fuzzy-vault was introduced by Jules and Sudan [4] in 2002. Anybody who has majority set of data that are overlapped with vault data can reconstruct the data set and hence the key.

Following is a brief example of fuzzy-vault. Alice places a secret K in a vault and locks it using unordered set A . Bob uses an unordered set B to unlock the vault to access K . He will succeed if and only if B and A overlap substantially [6]. Figure 5 illustrates this example.

In biometric fuzzy-vault, feature points are the data set. In the case of fingerprint biometric systems, data set can be minutiae points. Figure 6 illustrates minutiae points for fingerprints.

Uludag et al. [7] have implemented fuzzy-vault for fingerprint by hiding minutiae points in the vault. Their system's FAR reaches 0% but it has high time complexity.

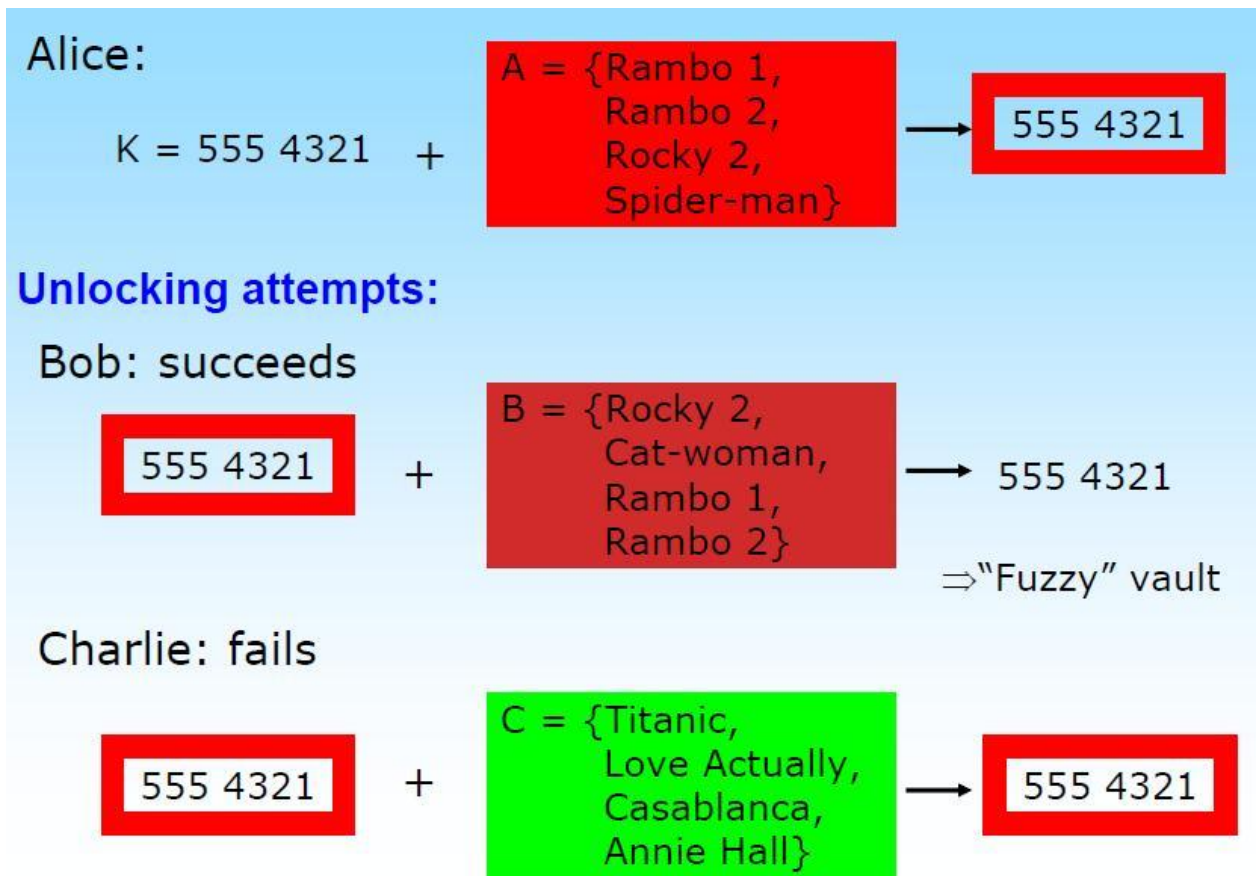


Figure 5: Fuzzy-vault example [6].

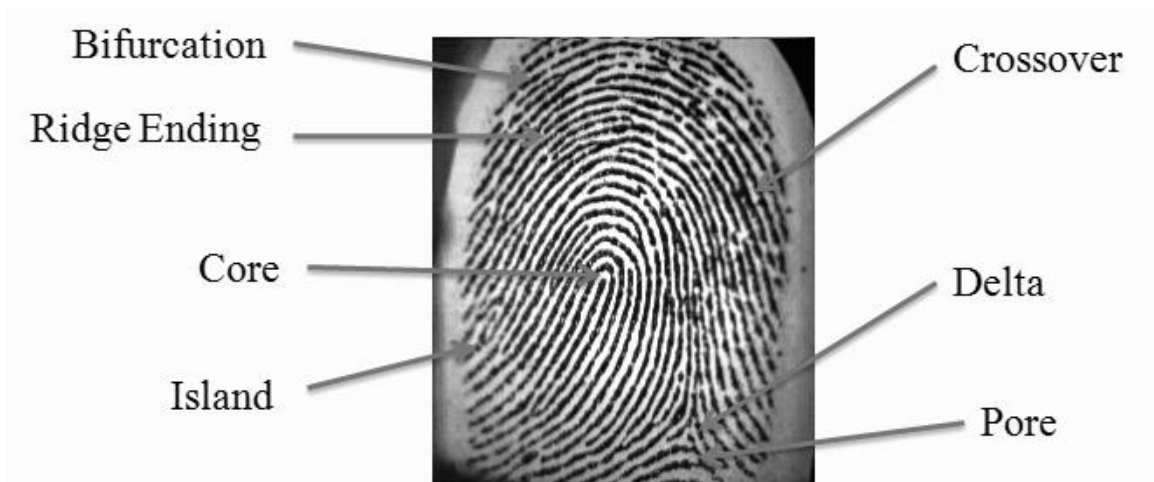


Figure 6: Fingerprint minutiae points.

Nandakumar et al. [8] have implemented a fingerprint based fuzzy-vault by proposing a new alignment technique for fingerprint to achieve high performance.

Nandakumar [9] has proposed a new method to secure iris code using fuzzy vault. This technique has been discussed in details in the following Section.

2.2.1 Fuzzy-vault for Iris

Iris is the angular region of the eye bounded by the pupil and sclera on either side. It is formed during fetal development and stabilizes during the first two years of life [10]. Each iris is distinctive even among identical twins. Figure 7 gives an example of iris showing its location in the eye.

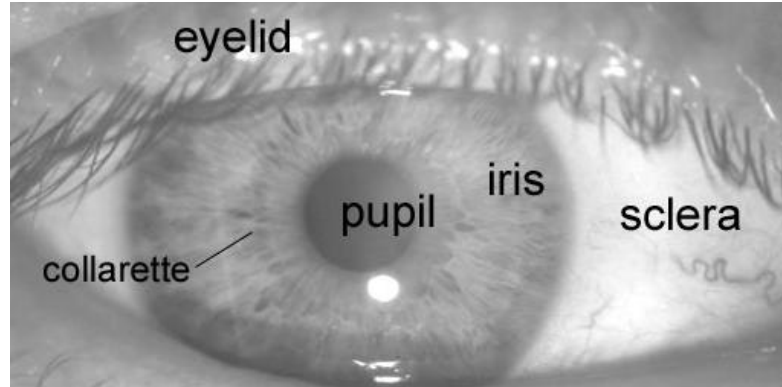


Figure 7: Iris and other eye components [10].

Daugman [11] has proposed a technique to recognize iris images by converting each iris to a fixed length binary vector. Multiple instances of the same iris should have approximately the same binary vectors. For more details, iris recognition has been discussed in details in Chapter 3.

Iris code is a fixed length binary vector and the relative order information between the bits is essential for matching. Because of this iris code cannot be secured directly using the fuzzy vault construct. A salt invariant transform function has to be applied first to the iris code. After that the resulting key can be secured using fuzzy-vault construct. This is what was proposed by Nandakumar [9]. Figure 8 illustrates fuzzy-vault encoding and decoding processes.

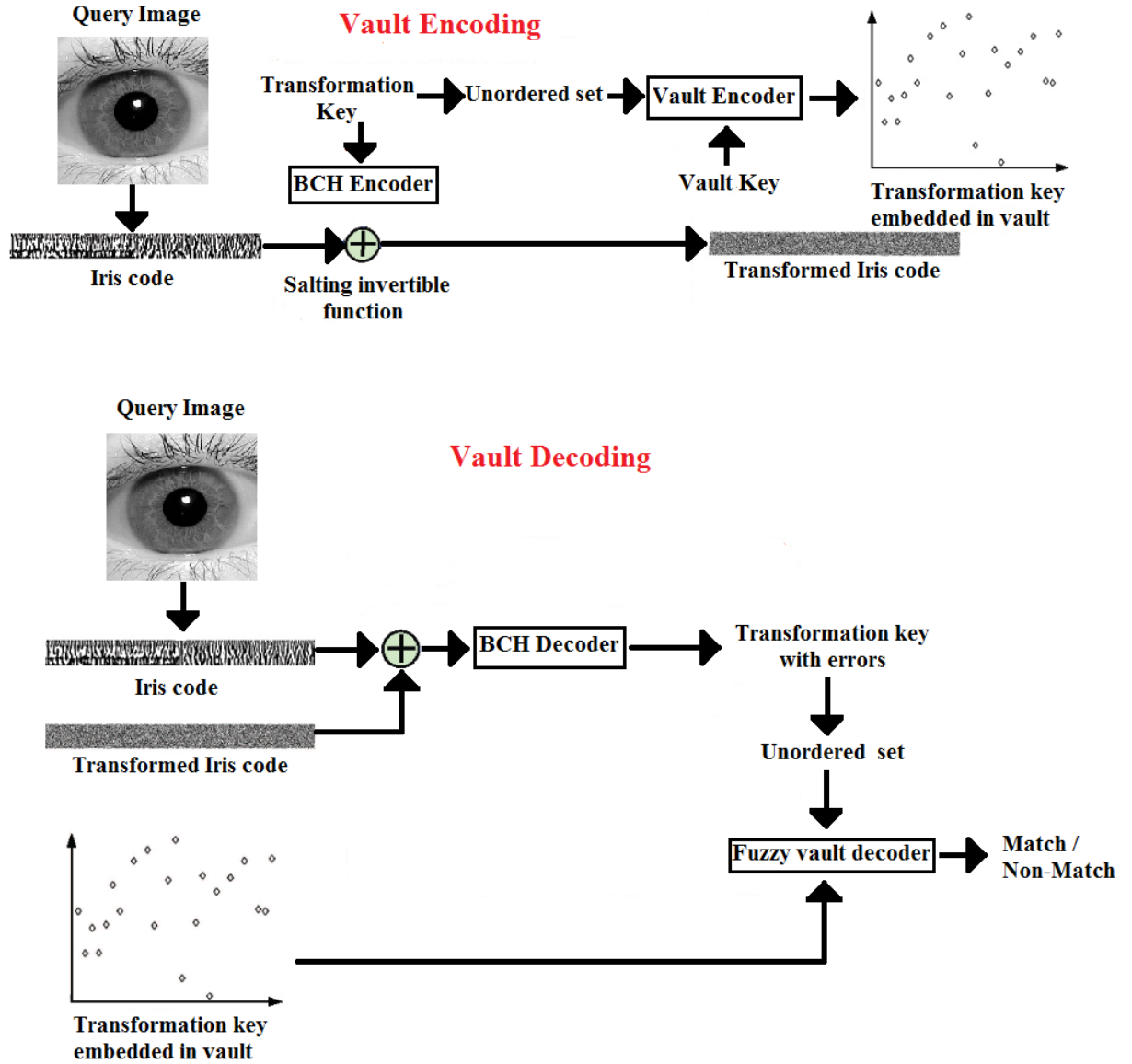


Figure 8: Vault encoding and decoding.

In vault-encoding process, two tasks need to be applied. In the first task a random key is generated and has been applied to the iris code as invertible transform function. It consists of BCH encoding to the transformation key and exclusive-or it with the iris code. In the second

task, the transformation key will be secured using fuzzy-vault construct. There will be a second key for securing the transformation and a set of random chaff points. Random chaff points are random points used to make the reconstruction of the second key infeasible. Chaff points play an important role in increasing the security of fuzzy-vault construct.

In vault-decoding process, inverse transform has been applied. It also consists of two tasks, exclusive-or between the transformed iris code and the query iris code. After that, BCH decoding will take place to produce the transformed key. Sometimes the resulting transformed key contains some errors because of intra-user variations. If the template and query iris code are sufficiently similar, then the recovered key should be sufficiently similar to the original key and the vault is decoded successfully. Fuzzy-vault for iris data has been implemented and discussed in details in Chapter 4.

2.3 Biometric Certification Techniques

Certification of biometric system represents developing a biometric system to store biometric templates on a separate media that certify the

association of contents like smart cards. Every enrolled person should have this card. In matching process data will be retrieved from this card. Unlike traditional biometric systems which retrieve data from an external database. A database may can be used, but not for matching. For example, new user wants to be registered; a preliminary check should be made to make sure that he/she did not already register.

Two famous certified biometric systems have been discussed here EyeCerts [12] and FaceCerts [13]. Each system is discussed in details in the following two Sections.

2.3.1 EyeCerts[®]

This system was developed by Schonberg and Kirovski [12]. EyeCerts is an offline system used to identify people by their irises. It offers a certified document that certifies the association of content on the document with iris feature. This system is highly cost effective and does not require high complexity. The system extracts and compresses the unique feature of a given iris using limited storage.

EyeCerts[®] is a printed document consists of two parts: arbitrary text and a barcode as shown in Figure 9. The barcode encodes the digital signature of iris feature and arbitrary text associated with the holder of the iris data. Iris feature is represented as a compressed image of the cardholder's iris.

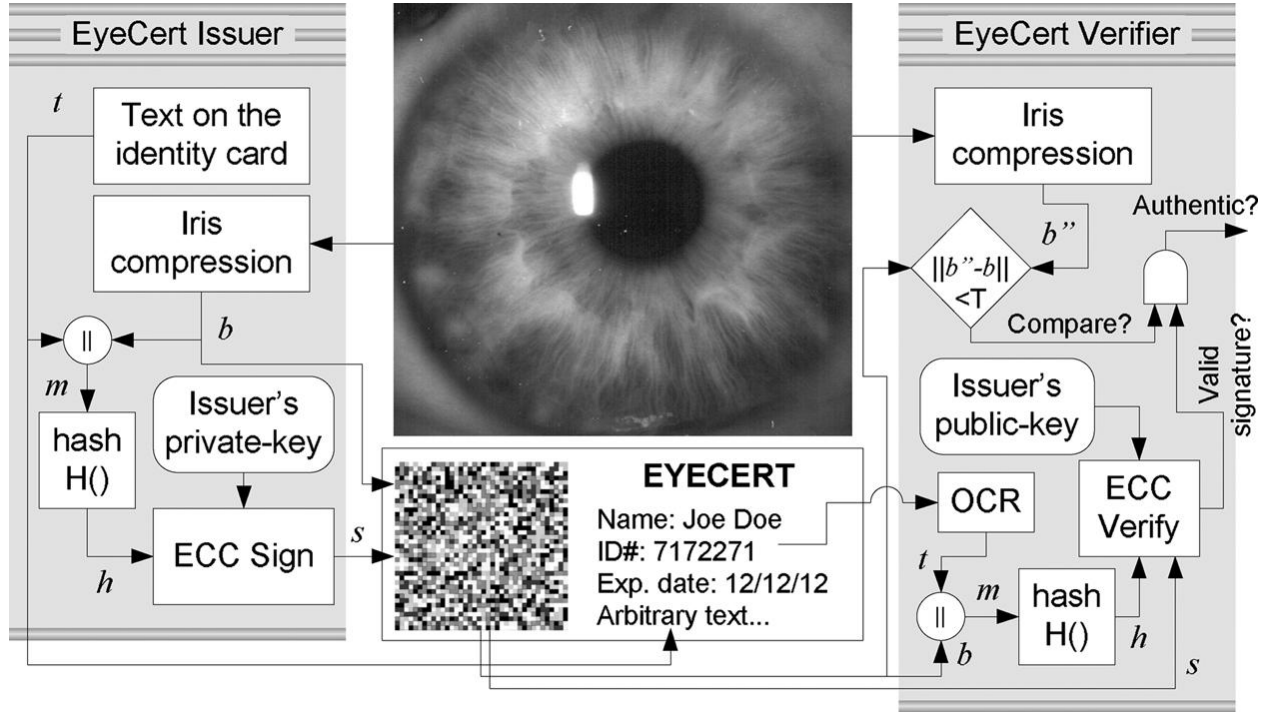


Figure 9: EyeCerts System [12].

As seen from Figure 9 that the system has two parts issuer and Verifier.

EyeCert[®] issuer scans the iris of human and compresses it. During iris capturing, identity text is entered to system t . Text t and compressed iris b will be concatenated together to produce a message m . A suitable hash function H is used to hash message m . Then, the result of the hash

function h will be signed with the issuer's private key producing s . Finally, the signature s and the iris data b will be encoded as a bar code and printed on the identity card.

EyeCert[®] Verifier scans the iris and compresses it b' . Simultaneously, the system will scan the barcode to extract iris data b'' and s'' and scan the printed identity text t'' . Iris data b'' and text t'' will be concatenated together to form message m'' . m'' will be hashed by using a hash function to produce h' . s'' will be decrypted by using the issuer's public key to produce h'' . A comparison will be made for h' and h'' and b' and b'' and according to distance threshold measure, authentication will be decided.

2.3.2 FaceCerts[®]

FaceCerts[®] has the same idea of EyeCerts[®] but it is for face biometric data. It was developed by Kirovski and Jojic [13]. It is simple, inexpensive and graphically secure identity system. FaceCerts[®] document has a printout of person's portrait photo, an arbitrary text, and a color barcode. Also it is an offline system and does not require high end printer technology. So because of this it does not require smart cards, all these

data can be printed on a paper, thus reducing cost. FaceCerts[®] is a good choice for driving license, passport, or visa identification. Figure 10 illustrates FaceCerts System.

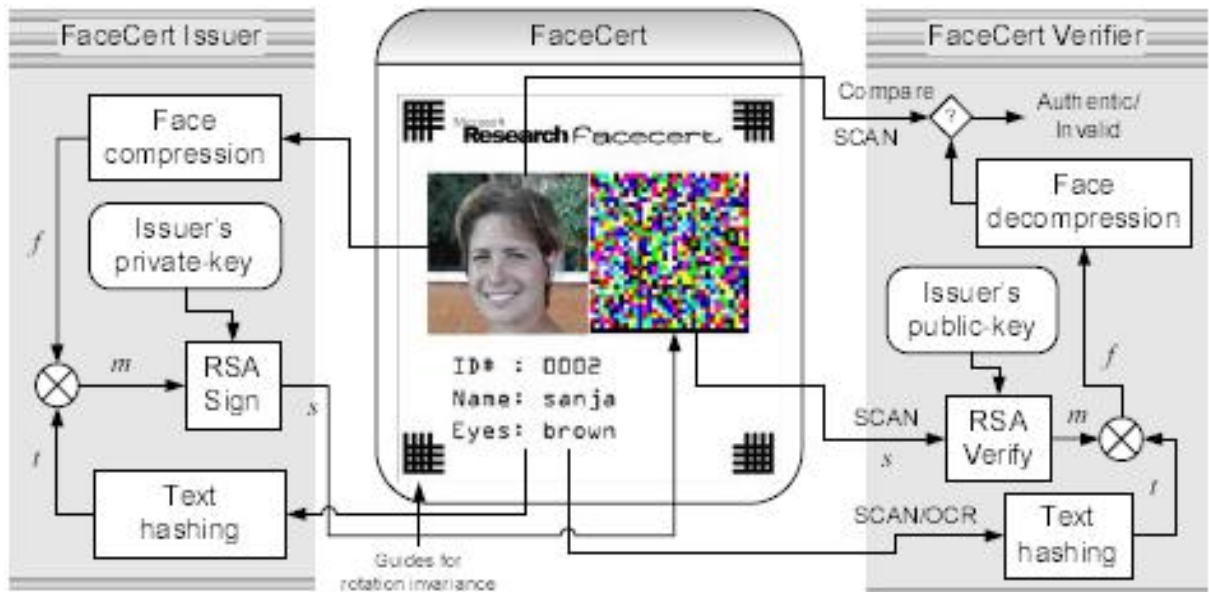


Figure 10: FaceCerts system [13].

As seen from Figure 10 that FaceCerts[®] document include: portrait photo of holder, arbitrary text, and colored bar code. Bar code encodes an RSA signature of the message hash and compressed representation of the face encompassed by the photo. From Figure 10 it is noticed that textual data pass a text hashing process. Textual data is hashed using a cryptographically secure hashing algorithm. The resulting hash is denoted

as t . Facial features on the photo are compressed using an algorithm that identifies the facial structure and compresses its features. A symbiotic eigenface - DCT based algorithm has been used for compressing the facial features.

Eigenfaces define eigen vectors of the set of faces that defines the face-space. They don't necessarily correspond to isolated features such as eyes, ears, and noses [14]. Eigenfaces formed by taking a set of images under the same lighting conditions and normalizing to line up the eyes and mouths. Figure 11 shows an example of eigenfaces.

After that eigenfaces can be extracted by using *Principled Component Analysis* (PCA). PCA is a data analysis technique that provides a roadmap for how to reduce a complex dataset to a lower dimension to reveal the hidden simplified structure that often underlie it [15].

Return back to Figure 10, after text hashing t and photo compression f , both are merged into a message m . After that m will be encrypted using RSA algorithm with issuer's private key to produce the barcode.

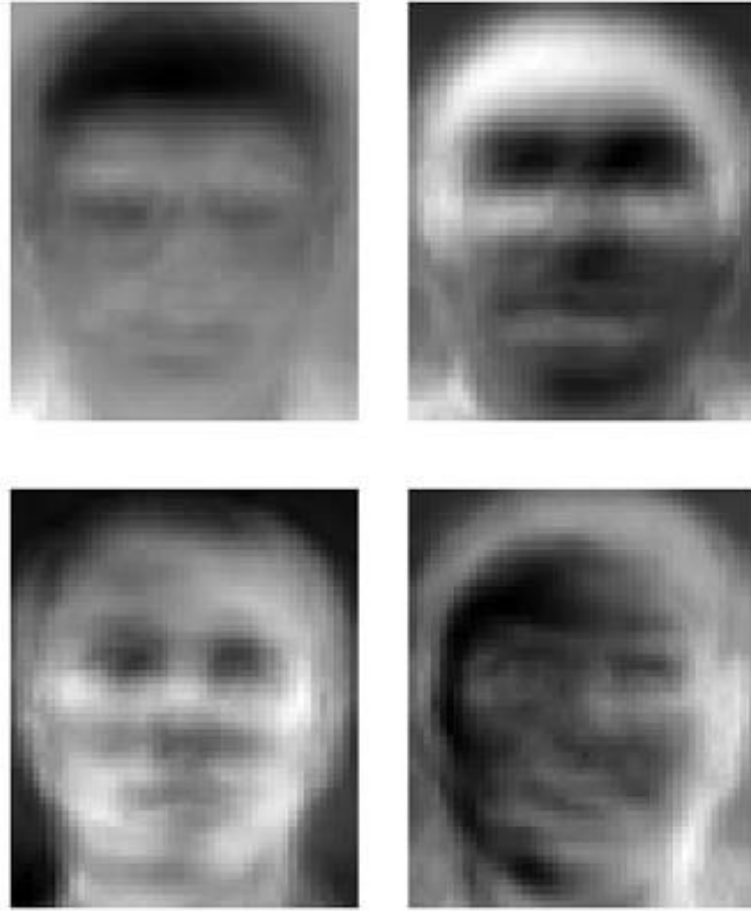


Figure 11: Eigenface example.

At verifier side, verifier will scan photo, barcode, and text. Text will be hashed to produce t' . Barcode is decrypted with the issuer's public key to produce message m' . Then from m' and t' , a compressed photo f' can be achieved. f' will be decompressed and compared with the scanned photo. After that, verifier can make decision for authentication.

2.4 Watermarking Techniques

As mentioned earlier, biometric systems have advantages over traditional personal identification techniques and the security of the systems data require careful treatment. Digital image watermarking can be used to achieve security of biometric data. Digital image watermarking is a technique used for authentication, tamper-proof digital content and protection of digital copyright. It is based on embedding specific information called watermark into host information or cover. Watermark can be company logo, copyright signature, etc.

Hembrooke [16, 17] has first described digital watermarking in his 1954 patent. It was a method for embedding an identification code into music fragments. This innovation idea is to improve the music ownership and intellectual property.

As discussed in Section 2.2, securing biometric templates through encryption is weak while the biometric data is decrypted. Therefore, biometric cryptography does not address the overall security of the biometric content. In this regard, digital watermarking can embed some information in biometric data, so it can provide security even for

decrypted data. In addition, this can provide a tracking mechanism for identifying the origin of the biometric data like FBI for example. On the other hand, watermarking can make the modification of the data by a pirate useless.

2.4.1 Watermarking Overview

Digital copyright protection is in a great demand as a measure for data security and protection. Digital watermarking has been developed to meet the needs of these growing concerns. Recently, it is witnessed a growing interest from research and commercial communities. Until recently, cryptography was deemed the “de-facto” solution for data security and protection. However, once the encrypted data is decrypted, it becomes an “open” platform for manipulations, altering and forgery. As a remedy to this deficiency, digital watermarking solutions are proposed instead since they can protect the protected content in “open” as well as in “closed” environments.

Moreover, digital watermarking technology is not new since it has been used in the paper industry for decades. For instance, any bank note holds a “semi-visible” paper watermark which becomes visible when the bank

note is held up to the light and looked at it. Figure 12 illustrates an example of such paper watermarks on a US bank note of US 20\$. In paper industry, the watermark directly is inserted into the paper during the papermaking process. This insertion makes it very hard to remove the watermark and therefore very hard to forge the bank note [18].



Figure 12: A sample bank note of 20 US \$ [18].

A digital watermark is defined as the extra information (or payload) that is imperceptibly and robustly embedded into the host data. A digital watermark typically contains information about the origin, status, or recipient of the host data [19]. Digital watermarks can be perceptible or

imperceptible. Imperceptible digital watermarks are more desired than perceptible watermarks in multimedia applications [19].

Digital watermarks can be classified according to their visibility into public (visible) or private (visible to authorized parties only). Private digital watermarks require special techniques to locate and extract. Unlike private watermarks, public one's can be seen by everyone without any extraction or location processes. Private digital watermarks should be located and extracted by authorized parties to protect host information against any type of alteration and/or modification [20].

2.4.1.1 Digital Watermarking Applications

Digital watermarking techniques can be used in (and for) many applications such as:

- 1- Copyright Watermarks or Copyright Protection: Used to embed creator/author information in the content [17].
- 2- Fingerprinting Watermarks: Used to enable tracking and tracing specific copies of the copyrighted contents.

- 3- Broadcast Watermarks: Used to protect media from being physically copied.
- 4- Annotation Watermarks: Provide a technique for embedding image metadata into the image itself.
- 5- Integrity Watermarks or Fraud Detection and Correction: Ensure the image authenticity or indicate any malicious content alteration [21].
- 6- Data Hiding Watermarks: Used to conceal hidden or secret information in an image (like *steganography*).
- 7- Bandwidth-Conserving Hybrid Transmission: Offers an opportunity to reuse and share existing spectrum to either backwards-compatibly to increase the capacity of an existing communication network such as legacy network [22, 23].

2.4.1.2 Digital Watermarking Systems

As mentioned in the previous Sections, digital watermarking represents a viable solution for copyright protection and tamper-proofing of digital content. Hartung [19] defined digital watermarking as embedding

information such as origin, destination, and access levels of multimedia data into the multimedia data itself.

The basic idea of a digital watermarking system is to add a watermark data to the host data to be watermarked. After receiving the watermarked host data by authorized parties, the embedded watermark payload can be recovered by using an appropriate decoding technique and proper decoding keys. Figure 13 gives the functionality of a typical digital watermarking system as described by Hartung [19].

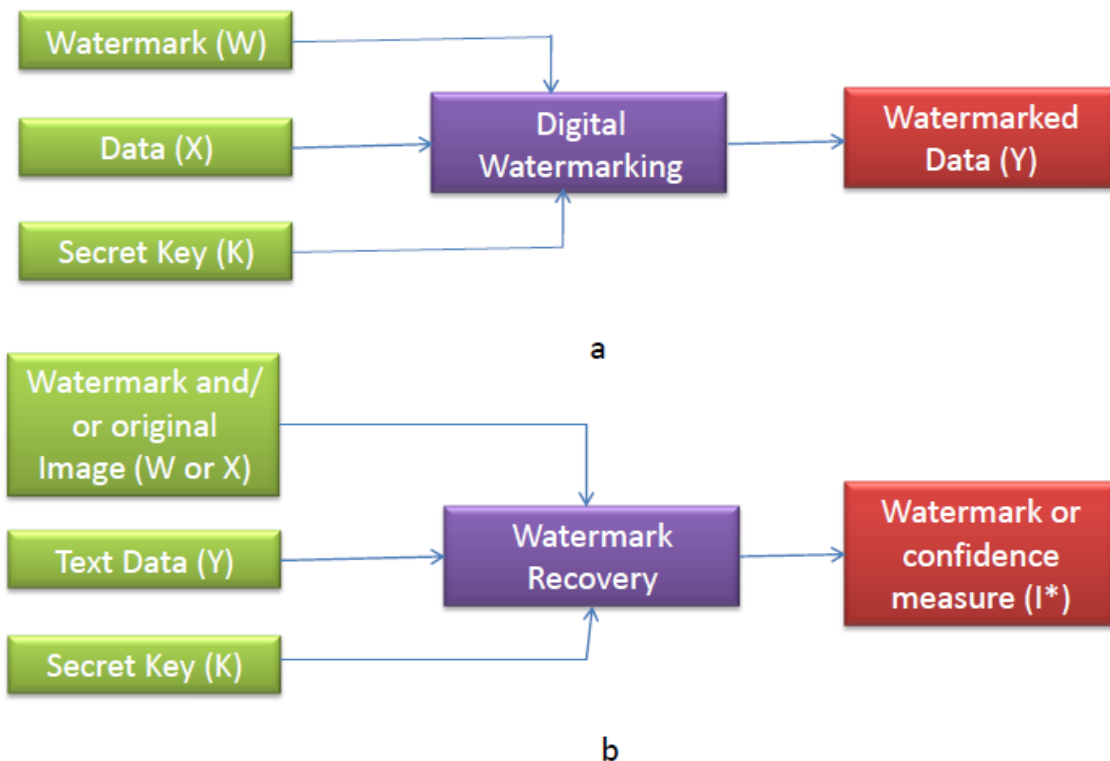


Figure 13: Typical digital watermarking system. a) Watermark encoding stage. b) Watermark decoding stage [19].

As shown from Figure 13, the watermark signal W , embedded into the host data X , can be a function of watermark information I and an embedding key K as defined below:

$$W = f_0(I, K) \quad (1)$$

Possibly, it may also depend on the host data X :

$$W = f_0(I, K, X) \quad (2)$$

The watermark information I is any information like the owner of the data and it can be used to increase the security of the entire system. The key also can be used to generate locations of altered signal components or the altered values.

After watermark creation, the watermark (W) is embedded into the host data to generate watermarked data (Y) such that:

$$Y = f_1(X, W) \quad (3)$$

The above operations are the watermark encoding. In watermark decoding, the embedded watermarked information or some confidence measure should be used to indicate the probability that a given watermark

is available in the test data. This is generated by using the original data as:

$$I^* = g(X, Y, K) \quad (4)$$

Or without the original data:

$$I^* = g(Y, K) \quad (5)$$

During watermark embedding, it is desired to keep the effects of watermark signal as imperceptible as possible in invisible watermarking applications. So the end user should not notice quality degradation in the signal due to watermarking. Because of this problem, some form of masking is generally used. For example, in audio watermarking, the frequency masking properties of the human auditory system (HAS) can be considered in designing the watermark. Also the masking effect of edges can be used in image watermarking. In the visible watermarking is the opposite case. No need to consider these masks.

Digital watermarking methods exist for almost all types of multimedia data, but they are much larger for image applications. They can be used to watermark images, audio, video, text, signals, and so on.

2.4.1.3 Classification of Watermarking Systems

Loo [24] has classified watermarking systems according to the following attributes:

- 1- Adaptability: Implies the consideration of an embedding model to control/adjust the watermark strength according to the strength of the host data. The most common models include the human visual/auditory systems (HVS/HAS) for image/video and audio digital watermarking systems, respectively.
- 2- Embedding Domain: The domain of the host data (host content) where the watermark payload is embedded. The embedding process can modify (alter) the attributes of the host signal which can be the coefficients of the host signal in the original domain (e.g., time or space) or in a “dual” transform domain (e.g., frequency or time-scale). Such transform domains can be implemented using the Fourier Cosine Transform (DCT), or the Discrete Wavelet Transforms (DWT).
- 3- Oblivious versus Non-oblivious: In oblivious digital watermarking systems, the watermark decoding (recovery) stage requires the

presence of the original host data for decoding purposes. However, in the case of non-oblivious systems, the original host content is not required. It should be noted that former systems are more robust to synchronization attacks.

- 4- Formation of the Composite Host Signal (Host Data): The watermark payload can be added to the host signal coefficients using an appropriate embedding algorithm. In non-additive watermarking systems, the content of the host content may undergo specific modifications to quantify the presence of the watermark payload.
- 5- Watermark Encoding: The watermark payload may require extra encoding to ensure robustness against transmission errors such as deletions and insertions. The watermarking strengthening process may include the incorporation of error control codes (ECCs) to mitigate the effects of the anticipated communication errors.
- 6- Watermark Robustness: Enables the watermark payload with resilience against intentional and accidental attacks. For instance, the design of the digital watermarking system should take into account robustness against unintentional manipulations such as

JPEG (image compression) or MPEG (video compression) and format conversions.

2.4.2 Watermarking for Biometric Systems

The applications of digital watermarking systems for securing and protecting biometric systems started to emerge recently. Ratha et al. [25] have proposed a data hiding method for fingerprint images in the wavelet scalar quantization (WSQ) compressed stream. The Discrete Wavelet Transform (DWT) coefficients are changed during WSQ encoding. This may cause possible image degradation and it should be taken into consideration. In Figure 14, original fingerprint image and watermarked fingerprint image are displayed. The second fingerprint has embedded data which is randomly generated bits and its size around 160-bytes. It is noticed in Figure 14 that the quality does not affected strictly.

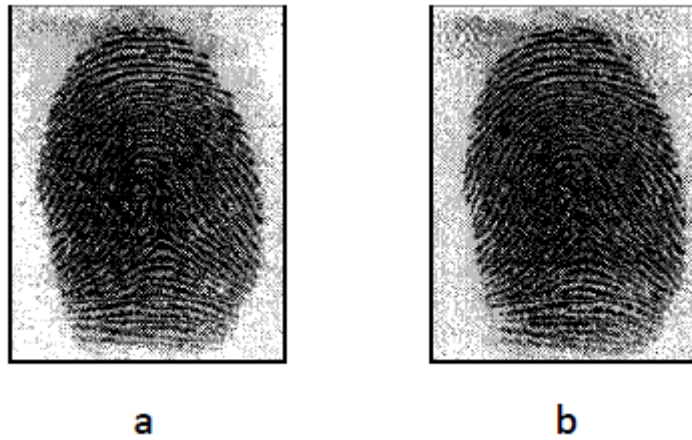


Figure 14: Compressed domain fingerprint watermarking a) Input fingerprint b) Fingerprint with data embedded [25].

Pankanti and Yeung [26] have proposed a watermarking technique for fingerprint image verification. Their technique was a fragile watermarking technique in which a spatial watermark is embedded in the spatial domain of a fingerprint image by utilizing a verification key. If any region of the watermarked image has been tampered, it can be localized directly. So this technique can be used to check the integrity of the fingerprint images. In Figure 15, the used watermark and the watermarked image are displayed.

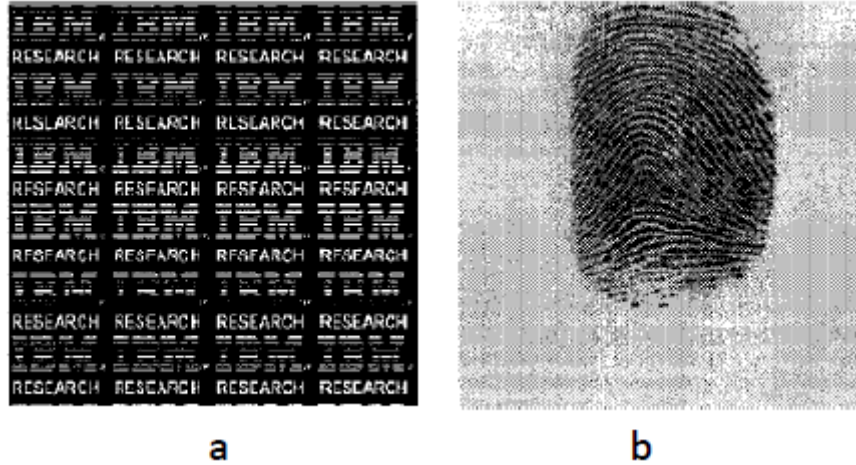


Figure 15: Fragile fingerprint watermarking a) Watermark b) Watermarked fingerprint image [26].

Pankanti and Yeung [26] technique does not lead to a significant performance loss in fingerprint verification. They proofed that by calculating the Receiver Operating Characteristics (ROC) curves on a database comprised of 1000 fingerprints. They have calculated (ROC) curves before and after the fingerprints were watermarked. They observed that curves are very close to each other which mean no significant performance loss.

Gunsel et al. [27] have described two techniques for watermarking fingerprint images. The first technique uses Gradient Orientation Analysis in watermarking embedding. Features that are extracted using gradient information will not be altered. The second technique keeps the

singular points in the fingerprint image, for example, into arch or right loop classes.

Low et al. [28] have presented a preliminary study on biometric watermarking. Their method is digitizing the offline handwritten signature into binary bit string as hidden biometric watermark. Three selected biometric watermarking techniques have been used. These techniques are Least Significant Bit (LSB) substitution, CDMA spread spectrum in spatial domain, and CDMA spread spectrum Discrete Wavelet Transform (DWT). Their performance criteria based on human visual inspection, Peak Signal to Noise Ratio (PSNR) and distortion rate. They have showed that CDMA spread spectrum in DWT is the best one in their results.

The same team (Low et al.) [29] have presented another biometric watermarking to embed handwritten signature in a host image as a notice of legitimate ownership. Their method's performance is validated against simulated frequency and geometric attacks, which include JPEG compression, low pass filtering, median filtering, noise addition, scaling, and rotation cropping. The results revealed that their method is able to

endure severe degradation on the host accuracy as their method showed remarkable robustness even if the host is strictly distorted.

Hong et al. [30] have presented a bit priority-based biometric watermarking. They stated that if the embedded biometric data are numeric, then the retrieval error will be very high. Their method is based on amplitude modulation and bit priority to embed high priority bits at good positions to reduce the retrieval error when they are converted to numeric data. Their experimental results showed a significant error reduction.

Jung et al. [31] have presented a method that identifies users at H.264 streaming by using fingerprints watermarking. Their system can survive under very low bit-rate compression. Fingerprint images are enhanced and a watermark is inserted with Discrete Wavelet Transform (DWT) technique. Their method achieved robust watermark extraction against H.264 compressed videos.

A robust fingerprint embedding scheme in the wavelet maxima points of fingerprint images is proposed by Ghouti [32]. In this scheme, minutiae points are not altered.

2.4.3 Detailed Biometric Watermarking Techniques

In this Section two techniques for watermarking biometric data have been discussed. Both techniques have the same data hiding method but they differ in the characteristics of the embedded data, the host image carrying that data, and the medium of data transfer. Both techniques were proposed by Uludag [33].

The first technique is a *steganography* based technique. In this technique, biometric features like fingerprint minutiae are hidden in a host image and transmitted via a non-secure communication channel. The function of the host image is just to carry data. This can be used for example, to transmit the fingerprint minutiae from any authorized verifier to the template database and vice versa. The security of this system is based on the secrecy of the communication channel. There is no relationship between the host image and the hidden data. Figure 16 illustrates this technique.

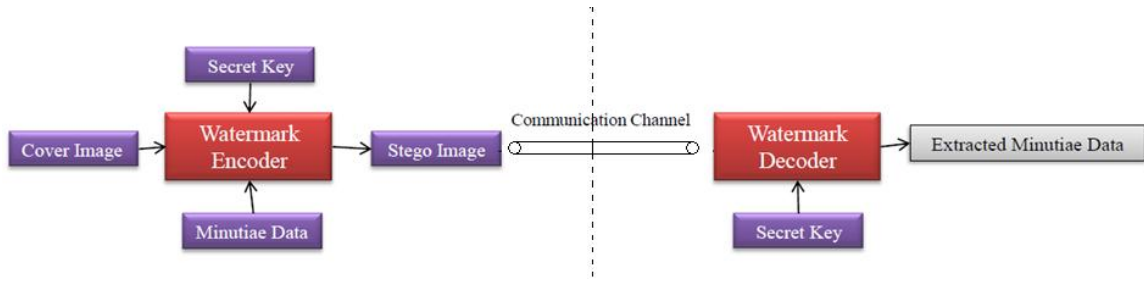


Figure 16: First technique for biometric watermarking [33].

One way for confusing the attacker, is to hide fingerprint minutiae in any synthetic fingerprint image. So the attacker will take this image and deals with it as a real fingerprint image without knowing that it carries minutiae data of another fingerprint. The security of this technique can be increased by encrypting the host image before transmission.

The second technique used to hide facial information such as eigenface coefficients into fingerprint images. Marked fingerprint image of a person can be stored in a smart card. Any person can use his own smart card in the verifier site. The fingerprint will be sensed and compared to the fingerprint stored on the smart card. After that facial information hidden in a fingerprint will be extracted to recover the face. The recovered face will be used as a second source for authentication either automatically or by a human in a supervised biometric application. Figure 17 illustrates the second technique for biometric watermarking.

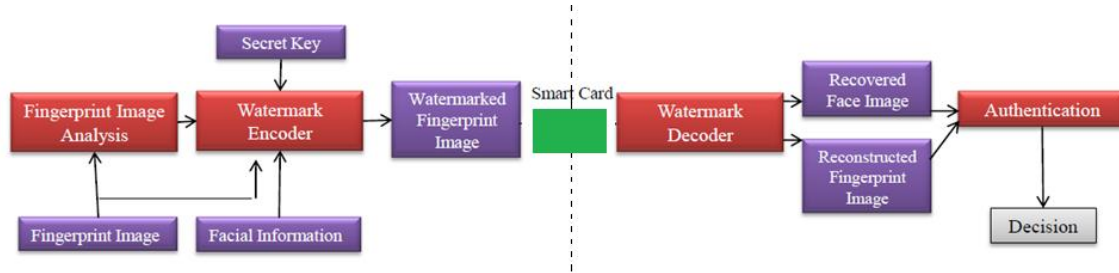


Figure 17: Second technique for biometric watermarking [33].

2.5 Comparative Study of the Existing Techniques

Techniques for securing biometric templates are compared according to the eight points of attack which illustrated at Figure 4. Table 1 shows this comparison.

Attacks	Fuzzy-vault	Biometric Certification	Biometric Watermarking
Fake biometric	Yes	Yes	Yes
Replay Data	Yes	No	No
Overriding (feature extractor)	Yes	No	Yes
Synthesizing	Yes	No	Yes
Overriding (matcher)	Yes	No	Yes
Intercept Channel	Yes	No	Yes
Modifying Data	Yes	No	No
Synthesizing	Yes	No	Yes

Table 1: Comparison for biometric template security techniques.

From Table 1, it is noticed that at point 1 this attack can be applied to all techniques because it depends on the biometric sensor or scanner more than the biometric data.

At point 2, biometric certification technique and watermarking technique do not have replay data attack. This is because the biometric certification technique is an offline system. So there is no external communication channel and nobody can access this system from external. For watermarking technique, any modification happen to the watermarked image will be exposed because watermarking offers a tamper-proof mechanism.

At point 3, 4, 5, and 6, just the certification of biometric system technique is safe from this attack because the system is offline.

At point 7, fuzzy-vault technique can just be attacked and modified if the attacker. For the certification of biometric system technique, there is no database to be modified. All the data encrypted and stored on the smart card. For watermarking, any modification happen to the watermarked image it will be exposed and this is one of the advantages of watermarking.

At point 8, certification of biometric system still safe from synthesizing because it is offline but the other schemes are not.

CHAPTER 3

MATHEMATICAL PRELIMINARIES

In this Chapter, mathematical and technical preliminaries, required for the development of the proposed techniques, are outlined. First, transform-based iris representation is explained since it is used for the development of the fuzzy-vault for securing iris templates. Then, the discrete wavelet transform is discussed given that it constitutes the foundation for the edge process (EP) model and the watermark embedding process. Since the EP process is based on statistical modeling of the wavelet coefficients, two well-known statistical wavelet modeling techniques are briefly reviewed; the Gaussian Mixture Model (GMM) and the Generalized Gaussian Distribution (GDD). While the GMM is used in the EP model to classify edges from background information, the GDD is used for the watermark extraction process. Finally, the Chapter concludes with some applications of the EP model on biometric raw data

to demonstrate its ability to effectively capture image edges in the wavelet domain.

3.1 Iris Representation

Iris is the angular region of the eye that is bounded by the pupil and sclera on either side. Figure 7 shows the iris and eye components. Daugman [11], among others, has proposed an efficient compact iris representation based on the Gabor wavelet. Although many systems have been developed for iris recognition, Daugman's system is dominantly used in most available commercial iris recognition systems.

To obtain a compact iris representation, the iris image undergoes many processing tasks. These tasks are: segmentation, normalization, and feature encoding. Each task is explained in detail in the following Sections.

3.1.1 Iris Segmentation

The segmentation phase is used to isolate the iris region from the other parts in the eye image. Eye image quality plays an important role in the

successful implementation of segmentation. Segmentation can be done using Hough transform [52]. Hough transform is used to determine the parameters of simple geometric objects such as lines and circles. Therefore, it is a good approach to deduce the radius and centre coordinates of the pupil and iris regions.

Segmentation is carried out as follows. First, an edge map of the iris image is generated by calculating the first derivatives of intensity values and apply a threshold on the resulting edge intensities. Second, circles are determined by considering the connected line passing through each edge point. The radius r and centre coordinates (x, y) define a circle according to the following equation:

$$x^2 + y^2 - r = 0 \quad (6)$$

Figure 18 illustrates an eye image and its edge maps resulting from Hough transform.

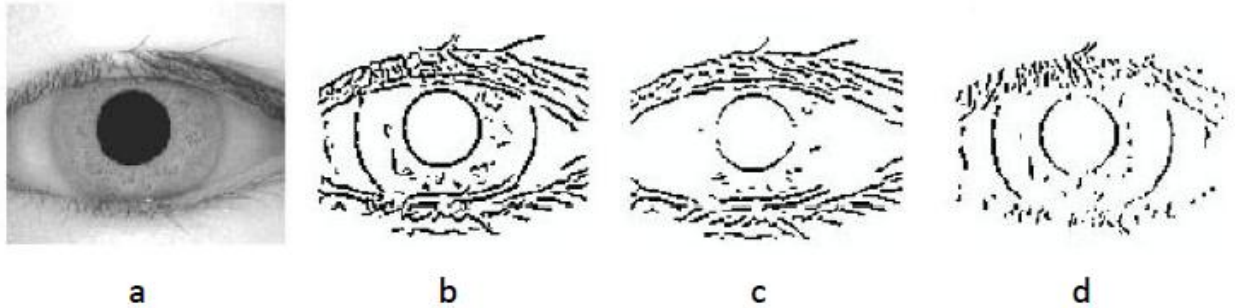


Figure 18: a) iris image b) edge map c) horizontal edge map d) vertical edge map [10].

An important point in segmenting the iris image is that eyelids may affect the circular shape of the extracted iris. Wildes et al. [34] have used the parabolic Hough transform to circumvent this problem. Parabolic arcs are used to approximate the upper and lower eyelids. As indicate in Figure 18, the eyelids are usually horizontally aligned. Taking the vertical gradients for locating the iris boundary is going to reduce the influence of the eyelids. Figure 19 shows an iris image after segmentation using the Hough transform as implemented by Masek [10].

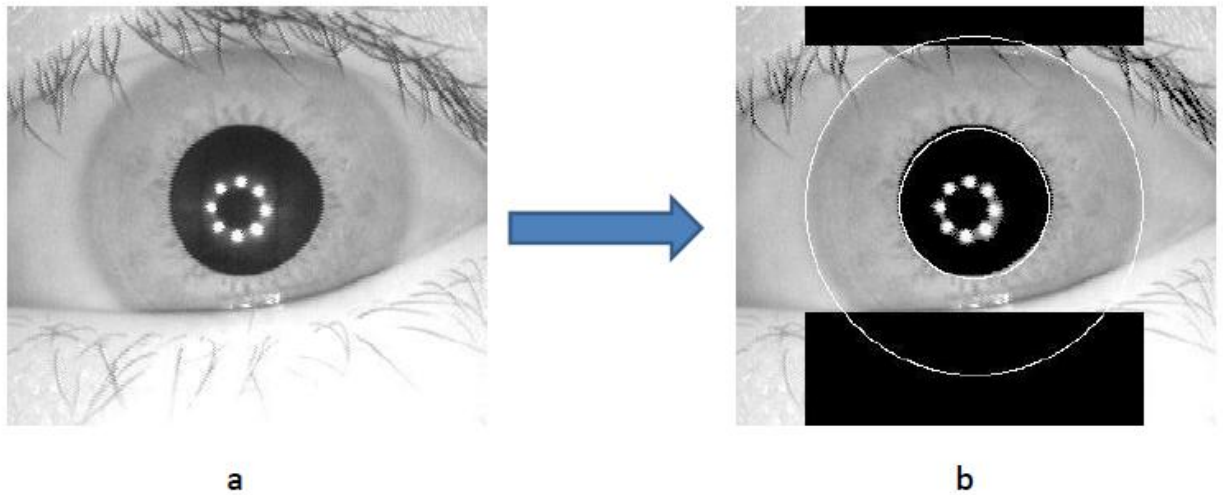


Figure 19: a) Iris image. b) Segmented iris image.

3.1.2 Iris Normalization

After the iris region is successfully segmented, it has a shape of a doughnut. It should be transformed to fixed dimensions so that it would be easier for representation. It should be noted that there may be dimensional inconsistencies between eye images due to the stretching of the iris caused by pupil dilation, varying image distance, rotation of the camera, head tilt, and rotation of the eye. The normalization process results in iris regions with similar dimensions.

The first step in normalization is to convert the polar coordinates of iris region to the Cartesian coordinates. Daugman [11] proposed a conversion process called the rubber sheet model. In this model, circled iris are

converted to a rectangular representation by transforming the Cartesian coordinates to a polar representation. The coordinate conversion is given by [11]:

$$I(x(r, \Theta), y(r, \Theta)) \rightarrow I(r, \Theta) \quad (7)$$

with

$$x(r, \Theta) = (1 - r)x_p(\Theta) + rx_l(\Theta) \quad (8)$$

$$y(r, \Theta) = (1 - r)y_p(\Theta) + ry_l(\Theta) \quad (9)$$

where $I(x, y)$ is the iris region image, (x, y) are the original Cartesian coordinates, (r, Θ) are the corresponding normalized polar coordinates, and x_p, y_p and x_l, y_l are the coordinates of the pupil and iris boundaries along the Θ direction.

The rubber sheet model takes into consideration pupil dilation and size inconsistencies in order to produce normalized iris with constant dimensions. The pupil center is going to be the reference point. Radial vectors pass through the iris region and the number of data points selected along each radial line is going to be the radial resolution. The

number of radial lines going around the iris region is called the angular resolution.

Sometimes, the pupil can be non-concentric to the iris circle. A remapping formula should be used to cover this problem. The remapping formula is needed to rescale points depending on the angle around the circle. Equation (10) gives the remapping formula.

$$r' = \sqrt{\alpha} \beta + \sqrt{\alpha \beta^2 - \alpha r_I^2} \quad (10)$$

where

$$\alpha = x_c^2 + y_c^2 \quad (11)$$

and

$$\beta = \cos \left(\pi - \arctan \left(\frac{x_c}{y_c} \right) - \Theta \right) \quad (12)$$

where x_c and y_c are the centre coordinates of the pupil, r' is the distance between the edge of the pupil and the edge of the iris at an angle Θ around the region, and r_I is the radius of the iris. Figure 20 illustrates the remapping formula.

For each radial line, a constant number of points are chosen. These points are taken irrespective of how narrow or wide the radius of a particular angle is. Normalization produces a 2D array with horizontal dimensions of angular resolution and vertical dimensions of radial resolution.

It was noticed in the segmentation process that eyelids can affect the iris circle. Hence, non-iris regions are removed from the normalized representation. Data points that occur along the pupil border or the iris border are going to be discarded. Figure 21 shows the iris image after normalization.

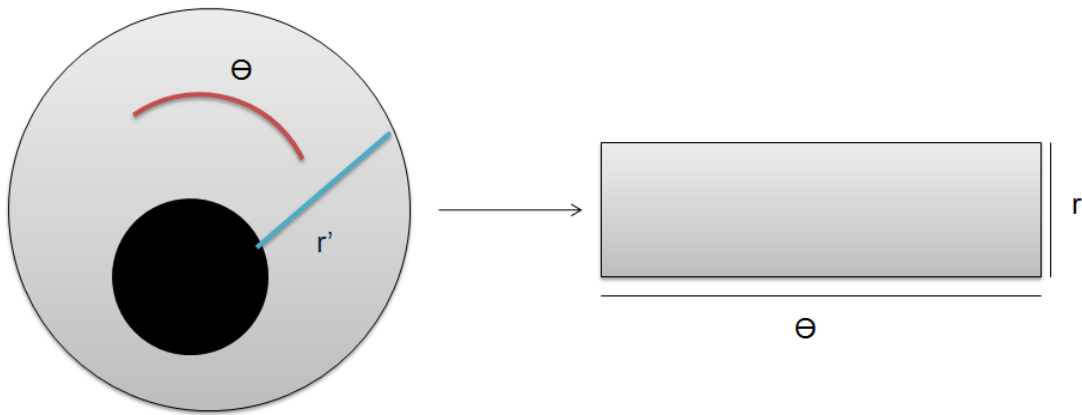


Figure 20: The remapping formula [10].

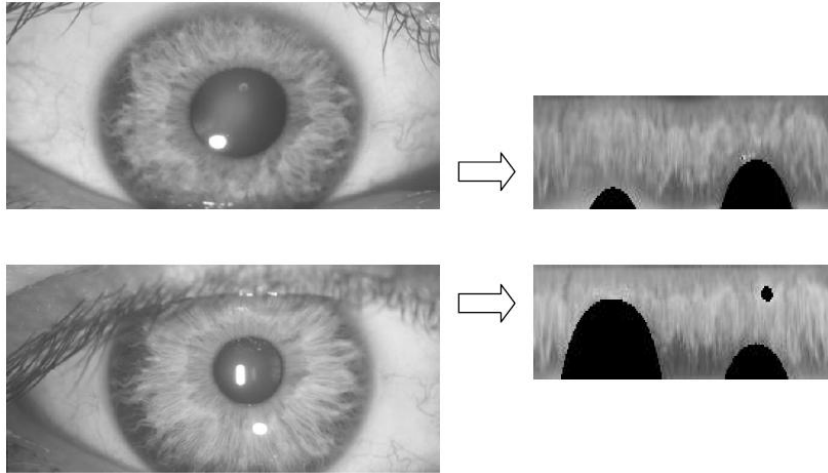


Figure 21: Iris normalization [10].

3.1.3 Feature Encoding

After the iris region has been successfully segmented and normalized, it needs to be encoded into a set of binary bits for storage and matching. Feature encoding means that the most discriminating information in iris pattern is going to be encoded to produce the iris code. Matching between iris images is based on these codes.

Gabor filters are used to extract the features from iris images. They provide optimum joint representation of a signal in spatial and frequency domains. A Gabor filter can be constructed by modulating a sine/cosine wave with a Gaussian. Sine wave is localized in frequency domain but not in time domain. Modulation of the sine with a Gaussian provides

localization in time domain but with loss of localization in frequency domain.

The signal is decomposed using a quadrature pair of Gabor filters, with the real part specified by a cosine modulated by a Gaussian and the imaginary part specified by a sine modulated by a Gaussian.

The bandwidth of the filter is specified by the width of the Gaussian. The centre frequency of the filter is specified by the frequency of the sine/cosine wave.

Daugman [11] has used a 2D version of Gabor filters to encode the iris data pattern. A 2D Gabor filter for an image domain (x, y) can be represented as follows:

$$G(x, y) = e^{-\pi[(x-x_0)^2/\alpha^2 + (y-y_0)^2/\beta^2]} \cdot e^{-2\pi[u_0(x-x_0) + v_0(y-y_0)]} \quad (13)$$

where (x_0, y_0) specify the position in the image, (α, β) specify the effective width and length, and (u_0, v_0) specify modulation. The real and imaginary parts of 2D Gabor filters are show in Figure 22.

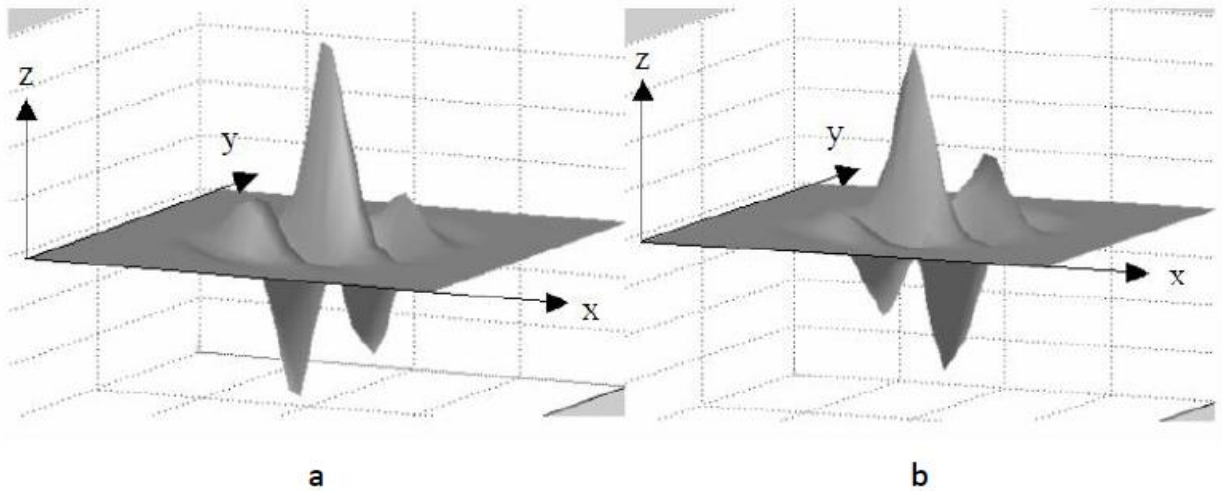


Figure 22: a) Real part characterized by a cosine modulated by a Gaussian b) Imaginary part characterized by a sine modulated by a Gaussian [10].

The output of the Gabor filters can be demodulated by quantizing the phase information into four levels for each possible quadrant in the complex plane. This demodulation is used to compress the data. Phase information provides the most significant information within an image. So taking the phase information is going to be used to encode iris information.

The four levels are represented using two bits of data. A total 2048 bits are calculated for the template. This creates a 256-byte iris template that can be used for efficient storage and comparison. Figure 23 illustrates the process of feature encoding using 1D Gabor filters using Masek's implementation of Daugman algorithm [10].

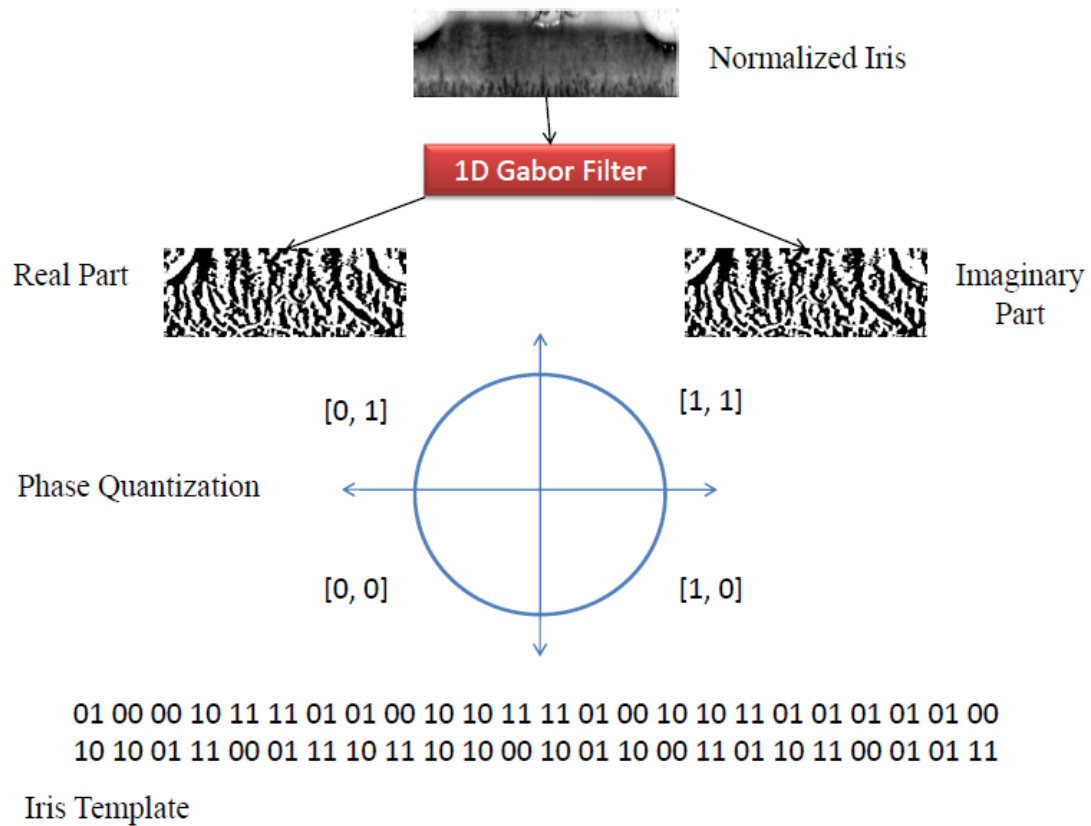


Figure 23: Feature encoding using 1D Gabor filters [10].

3.2 Wavelet Transform

Digital watermarking can be classified according to the embedding domain as explained in Section 2.4.1.3. Watermark signals can be embedded either in the spatial (time) or transform domain. Generally, transform domain always provides better robustness against attacks. The embedded watermark will be less perceptible in transform domain due to

the spread of the watermark signal over many spatial frequencies and better modeling of human visual system (HVS) [33].

There are two main types of transforms: Fourier and Wavelet transforms. Both transforms are introduced in the next section where the major differences between them are highlighted.

3.2.1 Wavelet and Fourier Transforms

The Fourier transform (FT) represents the process of transforming a signal from the time (spatial) domain to the frequency domain. On the other hand, the Wavelet transform (WT) is based on small waves called wavelets of varying frequency and limited duration. The WT captures not only frequency contents, but also temporal (spatial) contents like time (space) at which these frequencies occur. Both transforms can capture frequency which is a good similarity. However, there are dissimilarities. WT uses wavelet functions that are localized in space. This means that data are made sparse when they are transformed into the WT domain. The FT does not have this property. This sparseness can be utilized in a number of useful applications such as data compression, detecting features in images, and noise removal from time series [35].

Figure 24 illustrates a Windowed Fourier Transform (WFT). The window is simply a square wave. This window truncates the sine or cosine functions to fit a window of specific width. A single window is used for all frequencies in the WFT, so that the resolution of the analysis is the same at all locations in the time frequency plane.

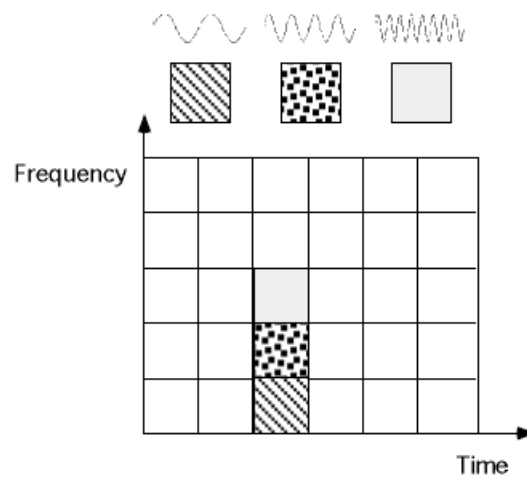


Figure 24: Windowed Fourier Transform (WFT) window [35].

In the WT, the windows vary. The WT has short-high frequency basis functions. This helps to obtain a detailed frequency analysis and isolate signal discontinuities. Figure 25 illustrates Daubechies wavelet in the time-frequency plane.

Also, the WT has an infinite set of possible basis functions unlike the FT which has just a single set of basis functions that utilizes sine and cosine

functions. This results in a number of different wavelet families. For each family, there are subclasses which can be distinguished by the number of coefficients, the level of iteration, and the number of vanishing moments. Figure 26 gives some examples of wavelet families.

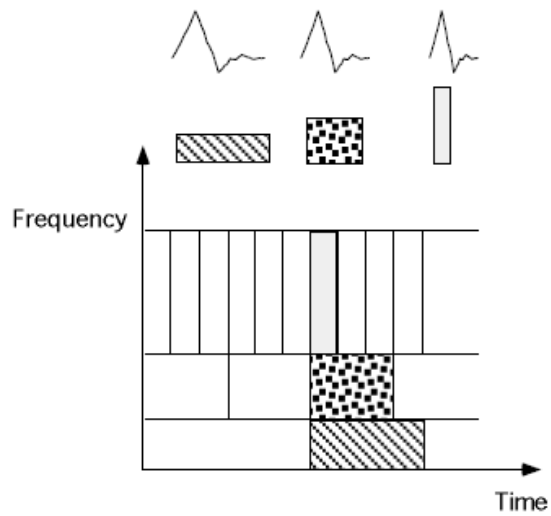


Figure 25: Daubechies Wavelet in time-frequency plane [35].

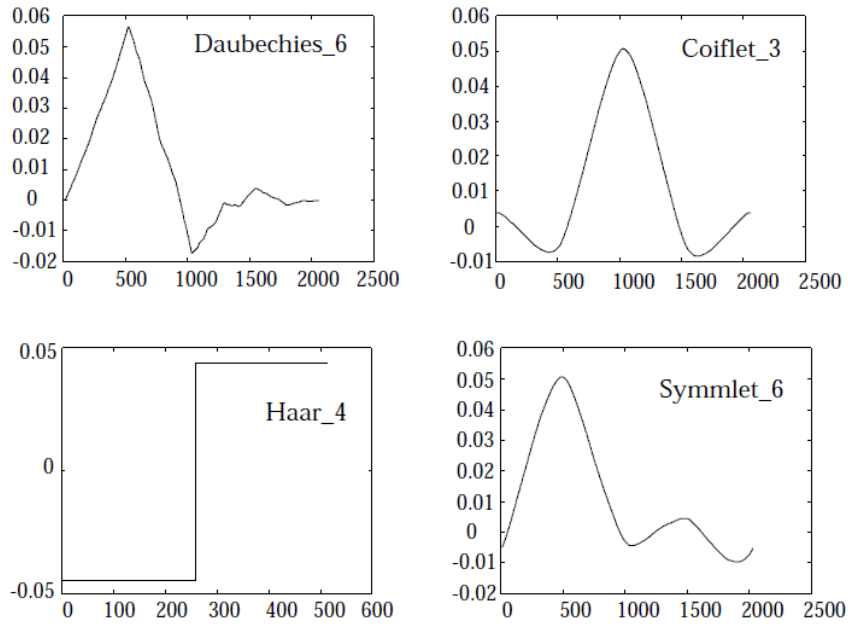


Figure 26: Different Wavelet families [35].

3.2.2 Data Analysis using Wavelet Transform

In the WT, $\Psi(x)$ represents the mother function of the wavelet functions used in the iterative construction of the WT. In fact, $\Psi(x)$ defines an orthogonal basis given by:

$$\Psi_{(s,l)}(x) = 2^{-\frac{s}{2}} \Psi(2^{-s}x - l) \quad (14)$$

where s and l are integers that scale and dilate the mother function $\Psi(x)$ to generate the wavelet functions. The scale index s indicates the wavelet width and l is a location index which gives its position. The mother

function, $\Psi(x)$, can be rescaled or dilated by powers of two and translated by integers. This means that once the mother function is known, everything about the basis functions can be derived as there are similarities between wavelet bases. These similarities are caused by the scales and dilations.

The analysis of the domain of the data being analyzed can be spanned at different resolutions. The scaling function is given by:

$$W(x) = \sum_{k=-1}^{N-2} (-1)^k c_{k+1} \Psi(2x+k) \quad (15)$$

$W(x)$ is the scaling function for the mother function $\Psi(x)$, and c_k are the wavelet coefficients. Wavelet coefficients must satisfy linear and quadratic constraints of the form:

$$\sum_{k=0}^{N-1} c_k = 2, \text{ and } \sum_{k=0}^{N-2} c_k c_{k+2l} = 2\zeta_{l,0} \quad (16)$$

where ζ is the delta function and l is the location index.

The WT has continuous and discrete implementations. In this thesis, the discrete implementation given by the Discrete Wavelet Transform (DWT) is considered. However, other implementations such as the Fast

Wavelet Transform (FWT), Wavelet Packets (WP) and Adapted Waveforms exist [35].

3.2.3 Two-Dimensional Wavelet Transform

In two-dimensional (2D) WT, a 2D signal (such as a digital image) is decomposed into a set of band-limited components. These components are called subbands. Subbands can be used to reconstruct the input 2D signal without errors. Hence, a 2D WT can be a good technique for image compression. Subbands are generated by band pass filtering of the input image. The size of any subband is smaller than the original image by half. Sub-bands are down-sampled but without loss of information. Reconstruction of the image is accomplished by up-sampling, filtering and summing the individual sub-bands.

Figure 27 illustrates the process of the 2D WT. As illustrated, the input image is processed by two filters (low-pass and high-pass) in parallel. When the image passes through any of these filters, it will be down-sampled by half. The same process is repeated iteratively to achieve a specific decomposition level. For example, Figure 28 shows an example

of a simple image. After applying the 2D WT, four subbands have been generated as shown in Figure 29.

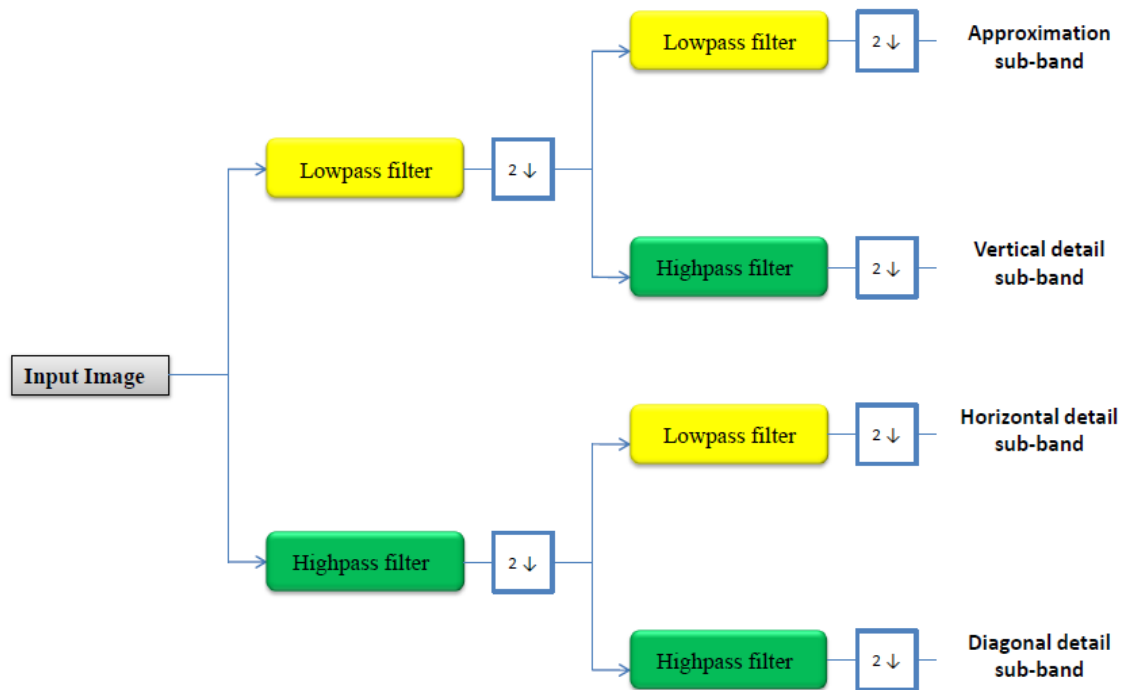


Figure 27: Wavelet Transform Process.



Figure 28: Example of a sample image.

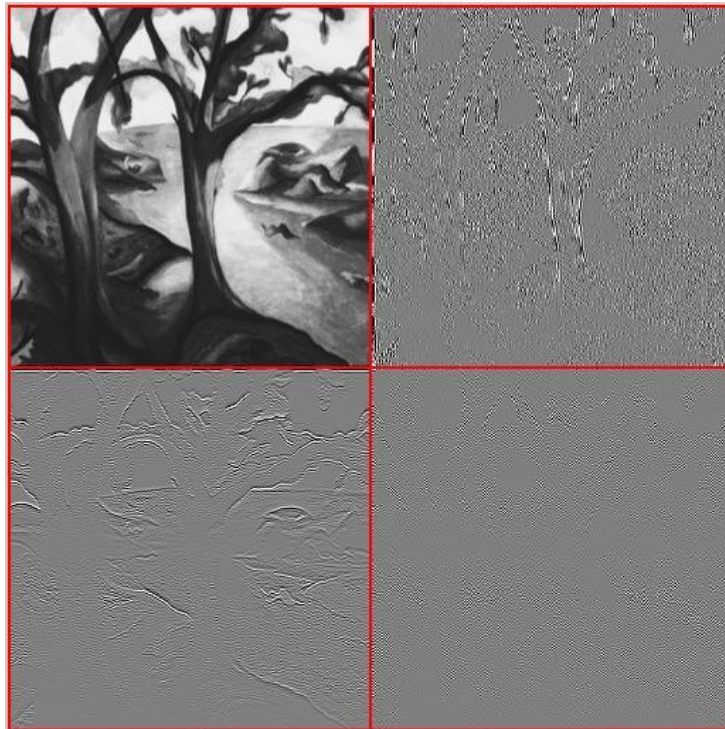


Figure 29: The Wavelet transform of the image in Figure 28.

Starting from the upper left quadrant in Figure 29 and moving clockwise, the four quadrants represent the approximation subband, the horizontal detail subband, the diagonal detail subband and the vertical detail subband, respectively. For image reconstruction, the inverse process is carried out by up-sampling, filtering and merging the sub-images until the image is reconstructed.

3.3 Wavelet Modeling

In this Section, two techniques for wavelet modeling have been demonstrated: the Gaussian Mixture Model and Generalized Gaussian Distribution.

3.3.1 Gaussian Mixture Model (GMM)

The Gaussian Mixture Model (GMM) is a probabilistic model for the estimation of the density. It is a model that consists of a number of Gaussian functions. These Gaussian functions are combined to provide a multimodal density. Component densities are linearly combined as follows:

$$p(x) = \sum_{j=1}^M p(x|j)P(j) \quad (17)$$

where $P(j)$ is the mixing parameter or the prior probability of the data points having been generated from component j of the mixture. All the priors are chosen to satisfy the conditions:

$$\sum_{j=1}^M P(j) = 1, \text{ and } 0 \leq P(j) \leq 1 \quad (18)$$

Then, the component density function for $p(x|j)$ can be normalized to become a class-conditioned density as follows [36]:

$$\int p(x|j)dx = 1 \quad (19)$$

Data points can be generated from the probability distribution by selecting a component j randomly with probability $P(j)$. After that, a data point is generated from the corresponding component density $p(x|j)$.

Individual component densities are given by Gaussian distribution functions. Gaussians have a covariance matrix which is a scalar multiple of the identity matrix. This shows that $\Sigma_j = \sigma_j^2 I$. Where I is the identity matrix and σ_j^2 is the variance of the component j . So, $p(x|j)$ is given by:

$$p(x | j) = \frac{1}{(2\pi\sigma_j^2)^{\frac{d}{2}}} e^{\left\{-\frac{|x-\mu_j|^2}{2\sigma_j^2}\right\}} \quad (20)$$

where μ_j is the mean value of density component j .

Many techniques have been developed for fitting GMM to a set of training data [36]. These techniques belong to the Maximum Likelihood (ML) family of algorithms which try to maximize the likelihood of the parameters for the given data set. There are many approaches for the ML-based algorithms including the negative log-likelihood, expectation maximization and stochastic estimation of parameters [36].

Several applications based on GMM modeling have emerged in various science and engineering fields. Raja et al. [37] use GMM to model the colors of an object to perform certain tasks such as real-time color tracking and segmentation. McKenna et al. [38] use GMM for on-line adaptation of models to cope with slowly-varying lighting conditions.

3.3.2 Generalized Gaussian Density (GGD)

Generalized Gaussian Density (GGD), also known as Generalized Gaussian Distribution, is a model that can be used to capture the global

behavior of the subband wavelet coefficients. Laplacian probability density function (PDF) is a special case of GGD when the shape parameter β is equal to 1. The GGD is defined by [39]:

$$p(x; \alpha, \beta) = \frac{\beta}{2\alpha\Gamma(1/\beta)} e^{-\left(|x|/\alpha\right)^\beta} \quad (21)$$

where $\Gamma(.)$ is the Gamma function ($\Gamma(z) = \int_0^\infty e^{-t} t^{z-1} dt$), $z > 0$, α is the

scale parameter, and β is the shape parameter. The scale parameter α models the width of the PDF peak. The shape parameter β is inversely proportional to the decreasing rate of the peak.

Do et al. [39] use GGD for texture retrieval in the wavelet domain. They combine two related tasks to GGD: feature extraction (FE) and similarity measurement (SM). Their method improves the texture retrieval rate from 65% to 77%.

Like GMM, the parameters of the GGD model can be estimated by using various techniques such as the ML, Moment Matching (MM), and Moment / Newton-Step (MNS). Varanasi and Aazhang [40] provide an investigation of the last three techniques where the ML algorithm is

found significantly superior for heavy-tailed distributions. It should be noted that the MM scheme works well if the region of the parameter space is in the vicinity of the Gaussian distribution. Finally, the MNS technique performs best for light-tailed distributions.

3.4 Edge Process (EP) Model

The Edge Process (EP) is a statistical model used to separate texture and edges regions from background or flat regions. It is mainly used for data compression and data hiding applications because edge regions are the best candidates for data hiding and textured regions are suitable for removal by compression without introducing noticeable distortions. The EP model has also many other applications.

Moulin and Mihçak [41] provide a theoretical analysis approach using the Estimation Quantization (EQ) technique [42] and the spike process model [43] to evaluate data hiding capacity of real images. Voloshynovskiy et al. [44] propose the Edge Process (EP) model. Data hiding estimates based on the EP process are compared to those obtained using the EQ and spike process models. Reported results show that the

EP process achieved more accurate estimates for data hiding capacities yielding improved performance in terms of robustness and imperceptibility. Other applications including image compression and denoising have been considered therein.

It should be noted that the most sophisticated statistical image models are implemented in the transform domain such as the Discrete Cosine Transform (DCT) or DWT domains. In the EP model, the DWT domain is used due to its relevance to the modeling of the subband wavelet coefficients. Also, the DWT has been preferred over the DCT for image coding applications in emerging standards such as JPEG 2000 image coding standard [52]. Moreover, the DWT provides a good energy compaction of the image being analyzed and it has desirable properties such as sparsity, locality and multi-resolution.

3.4.1 Applications of the EP Model

The EP model can be used for image regeneration using the image statistical descriptions; operational entropies (image compression); and image denoising. The entropy represents an average length of the code for lossless data representation. The EP model provides obvious

enhancement to image quality objectively and perceptually as compared to the EQ model. Moreover, the EP model achieves lower entropy as compared to the EQ model [45]. Many results demonstrate that the EP model yields better performance in reference applications [45].

3.4.2 Edge Process Analysis

There are various models to characterize dependencies between wavelet coefficients. These models can be categorized into three groups: *interscale* dependencies; *intrascale* dependencies; and both dependencies. The EP model addresses the intrascale dependencies since they are usually stronger than the interscale dependencies as reported by Liu and Moulin [46].

The GGD is the most widely used class of intrascale image models. The GGD model captures the global behavior of the wavelet coefficients. A significant gain can be achieved if the coefficients are considered being locally Gaussian rather than globally Laplacian. So, a mathematical relationship can be established between the local and global models. This relationship can be achieved using the infinite GMM model. In the EP model, the number of Gaussian channels is limited to K instead of an

infinite number. The image is split into K classes according to their variances of the wavelet subband coefficients.

The global Laplacian PDF, $p_x(x)$, can be obtained as a weighted mixture of zero-mean conditionally Gaussian PDFs, which is conditioned on the local variances σ_x^2 and exponential prior on σ_x^2 that capture the local image statistics:

$$p_x(x) = \int_0^{\infty} p_{x|\sum_x^2}(x|\sigma_x^2) p_{\sum_x^2}(\sigma_x^2) d\sigma_x^2 \quad (22)$$

where $p_{x|\sum_x^2}(x|\sigma_x^2) = (1/\sqrt{2\pi\sigma_x^2}) e^{-(x^2/2\sigma_x^2)}$, $p_{\sum_x^2}(\sigma_x^2) = \gamma_1 e^{-\gamma_1 \sigma_x^2}$, and

γ_1 is the scale parameter of Laplacian PDF. So, Equation (22) can be rewritten as:

$$p_x(x) = \int_0^{\infty} \frac{1}{\sqrt{2\pi\sigma_x^2}} e^{-\frac{x^2}{2\sigma_x^2}} \gamma_1 e^{-\gamma_1 \sigma_x^2} d\sigma_x^2 = \sqrt{\frac{\gamma_1}{2}} e^{-\sqrt{2\gamma_1}|x|} \quad (23)$$

This relationship provides a link between the global and local statistics of the image coefficients. Data is considered to be locally zero-mean Gaussian with the variance distributed according to the exponential PDF and having Laplacian global statistics at the same time.

3.4.3 Edge Process Definition

Consider an image Z with support L . Let M_1, M_2, \dots, M_n be the set partitioning L where all M_i are disjoint connected sets, $M_i \cap M_j = \emptyset$ for $i \neq j$ and $\bigcup_l M_l = L$. M_l is a region separated according to its variances as follows:

$$M_l = \{z_l : Z[l] \rightarrow \in (0, \sigma_{z_l}^2 [l])\} \quad (24)$$

All coefficients are approximated by a Gaussian distribution with different local variances $\sigma_{z_l}^2 [l]$. The EP model assumes two distinctive sets of coefficients in wavelet domain for each subband. One set belonging to the flat region and one set belonging to edges and texture regions. Propagating along the edge, any transition corresponding to an edge or to a fragment of texture consists of several distinct mean values.

Scalar Uniform Threshold Quantizer (UTQ) is used to segment regions. It is used for a given subband that is characterized by global GGD. It was assumed that the UTQ has uniformly spaced decision levels.

After segmentation, the EP model regions are defined as follows:

$$M_1 = \{z : Z[l] \rightarrow \in (0, \sigma_z^2 [l])\} \quad (25)$$

$$M_2 = \{z: Z[i] \rightarrow \in (\bar{z}_j[i], \sigma_{z_j}^2 [i])\} \quad (26)$$

where j is used to indicate the data behavior along the j th local edge, $M_1 \cup M_2 = L$ and L represents a particular subband. Region M_1 represents all flat regions within a subband and is assumed to be zero-mean Gaussian random variables with local variance $\sigma_z^2 [i]$. Region M_2 represents texture and edges regions. In region M_2 , each distinctive geometrical structure corresponding to edge or texture transition is decomposed into a set of local mean constellations. A particular mean value $z_j[i]$, $j= 1, \dots, J$ where J is the number of mean levels, propagates along the edge creating edge process. Edge parameters and orientations depend on the mutual orientation of the edge and of the subband. Usually transitions along the edge have longer stationary length than the transitions within the texture. This explains the existence of higher correlations along the edges. Moreover, the stationary condition is stricter for edges than for textures.

CHAPTER 4

PROPOSED TECHNIQUES

In this Chapter, the proposed techniques in this thesis are explained in details. First, a fuzzy-vault scheme for securing iris templates has been implemented as described by Nandakumar [9]. Then, a new watermarking technique for protecting fingerprint images using the edge process model has been implemented.

4.1 Fuzzy-vault for Iris Templates

In this Section fuzzy-vault for iris templates is going to be implemented as described by Section 2.2.1.

First, human iris needs to be recognized by computer system. This task has been accomplished through Masek's implementation [10] of Daugman's [11] algorithm, which has been described in details in Section

3.1. This algorithm takes an iris image and extracts the iris from other eye parts. After that the iris will be normalized resulting in an iris code of a specific size. Figure 30 illustrates two different iris codes generated by Masek's implementation.



Figure 30: Example of two different iris codes.

As stated earlier in Section 2.2.1, that the iris code is a fixed length binary vector and the relative order between the bits is extremely important for matching. Because of this, iris code cannot be secured directly with fuzzy-vault construct. To solve this problem, a salt invariant function needs to be applied first.

Fuzzy-vault has two phases (As seen previously in Figure 9): vault encoding and vault decoding. These phases are explained in details in the following Sections.

4.1.1 Vault Encoding

In this phase, there are two steps. In the first step, an invertible transformation function is applied to the iris code template based on a randomly generated transformation key. In the second step, the transformation key is presented as an unordered set and is secured using fuzzy vault construct. The vault encoding phase has been summarized in Figure 31.

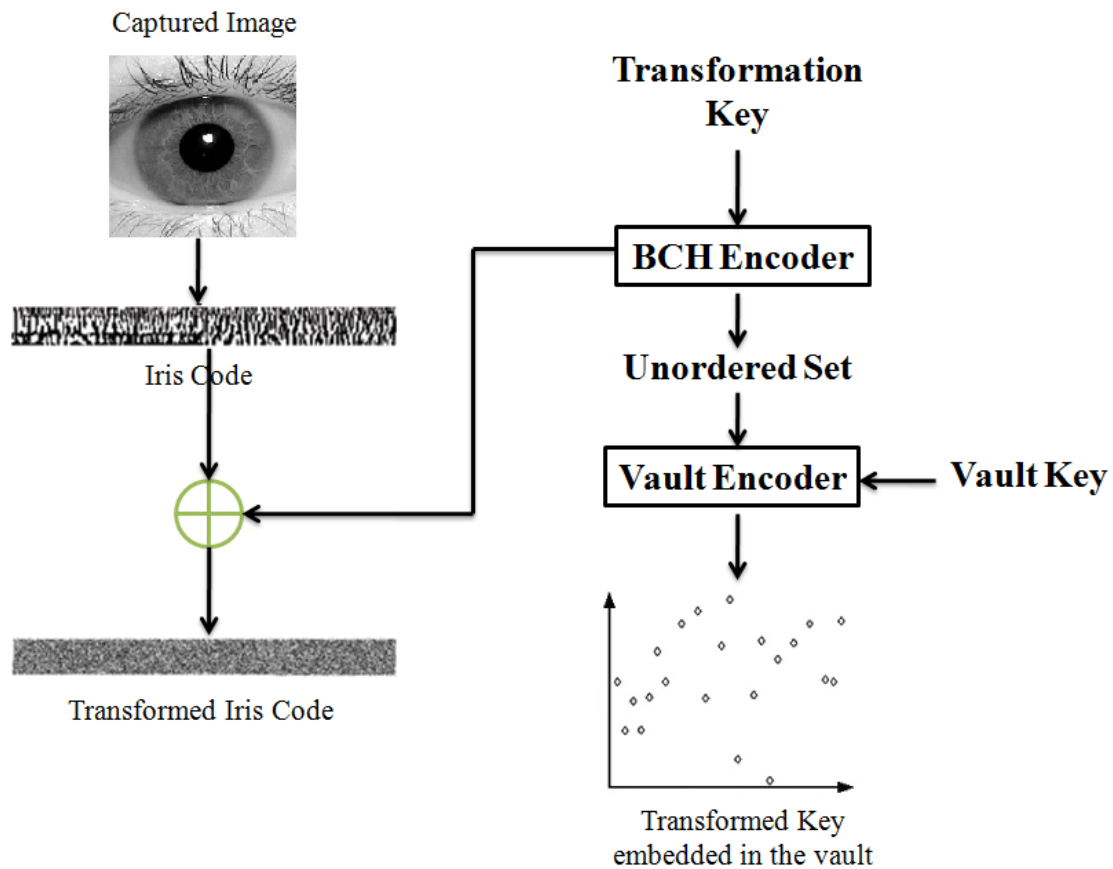


Figure 31: Vault encoding phase.

The first step consists of two operations BCH encoding and an exclusive-or operation. BCH encoding is a type of error-control codes (ECCs) that are used to increase the robustness of the transformed iris template. Let B be a BCH (Z_I, Z_K) encoding function, which takes a message M of length Z_K ($Z_K < Z_I$) and appends $(Z_I - Z_K)$ error correcting symbols to it in order to generate a codeword $T = B(M)$ of length Z_I . A primitive binary BCH encoding scheme has been employed, where Z_I is chosen to be $(2^j - 1)$ and j is an integer greater than or equal to 3.

Let T_T be an iris code template of length N_I bits that is to be secured using the fuzzy vault. Firstly, the template T_T is partitioned into r non-overlapping components where each component contains exactly Z_I bits. r is selected such that $rZ_I > N_I$. When $N_I < rZ_I$, an appropriate number (i.e. $(rZ_I - N_I)$) of zero bits are appended to the iris code template. Next, r vectors are randomly as M_1, M_2, \dots, M_r , each of length Z_K bits. These vectors constitute the transformation key K_I of length rZ_K bits. The BCH encoder B is applied individually to the binary vectors M_1, M_2, \dots, M_r to obtain the codewords $B(M_1), B(M_2), \dots, B(M_r)$. Finally, an exclusive-or operation is performed between the r codewords generated by the BCH encoder and the corresponding components of the iris code template to

obtain the components of the transformed iris code. The encoding phase can be represented as a function F_I that takes the iris code template T_T and the transformation key K_I as inputs and generates the transformed iris code T_* such that $T_* = F_I(T_T, K_I)$.

The key K_I has been secured using the fuzzy vault construct. Key recovery (decoding) has been designed in such a way that it does not require the relative order information between the components of key K_I . The components set of K_I and random chaff points will be encoded. A second key K_2 is used. K_2 is of length $16n$ bits where n is the degree of the encoding polynomial. A 16-bit CRC code is appended to the secret key K_2 in order to obtain a new secret key K_2' containing $16(n + 1)$ bits. The generator polynomial for CRC is $G(x) = x^{16} + x^{15} + x^2 + 1$. The secret key K_2' is encoded into a polynomial P of degree n in F by partitioning it into $(n + 1)16$ -bits values c_0, c_1, \dots, c_n and considering them as coefficients of P , i.e. $P(x) = c_n x^n + \dots + c_0$. Then, the polynomial P is evaluated at all points in key set K_I and chaff points. After that, the elements are randomly reordered to obtain the vault U . Only the vault U and transformed iris code T_* will be stored in the system. Figure 32 shows an example of encoded vault.

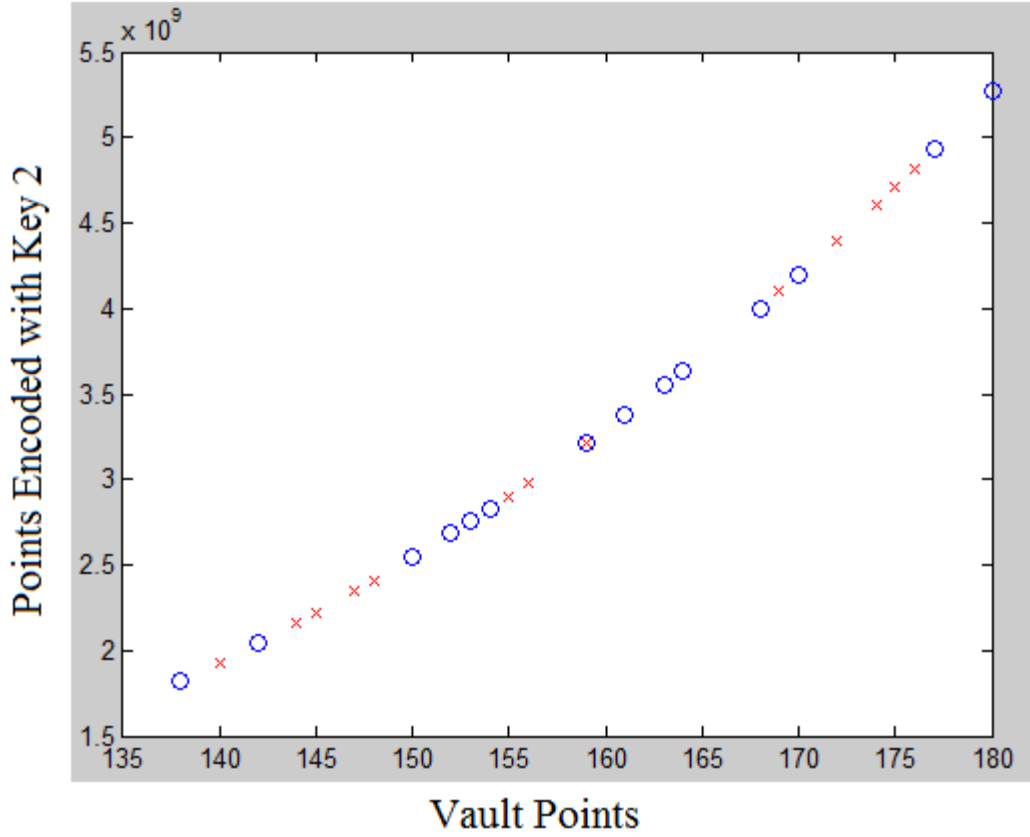


Figure 32: Example of encoded vault. The blue points are the key points and the red points are random chaff points.

4.1.2 Vault Decoding

Vault decoding phase or key recovery consists of two main steps. First, the inverse transform is applied to the transformed iris code template T_* using the query iris code T_Q . Since the template and query iris codes will not be identical due to intra-user variations, the recovered key K_I' may have some errors. Second, the transformation key K_I' is used to decode

the vault U . If the template and query iris codes are sufficiently similar, the recovered key K_I' will be sufficiently similar to K_I and the vault can be successfully decoded.

The inverse transform consists of two operations, an exclusive-or operation followed by BCH decoding. Let T_Q be the query iris code of length N_I bits. The query T_Q has been partitioned into r non-overlapping components and each component contains exactly Z_I bits. An exclusive-or operation is performed between the r components of the query iris code and the corresponding components of the transformed iris code T_* to obtain the corrupted code words. Let B^{-1} be a (Z_I, Z_K) primitive binary BCH decoding function that takes a corrupted codeword $B'(M)$ of length Z_I and decodes it into a message M' of length Z_K . If the Hamming distance between the corrupted codeword $B'(M)$ and the original codeword $B(M)$ is less than the error correcting capability of the BCH coding scheme, the decoded message M' would be the same as the original message M .

The corrupted codeword $B'(M)$ has been decoded using the BCH decoder to recover the components of the transformation key K_I' . Due to problems such as occlusion, there may be large differences between some of the

template and query iris code components and the corresponding components of the transformation key that cannot be recovered correctly.

To find the coefficients of a polynomial of degree n , $(n + 1)$ unique projections are necessary. A polynomial P^* is constructed by Lagrange interpolation. After that, the coefficients of the polynomial are concatenated to obtain a $16(n + 1)$ -bit string K_2^* and CRC error detection is applied to K_2^* . The vault decoding phase is summarized in Figure 33.

Fuzzy-vault is applied to one iris image from CASIA database [47] and is decoded in order to be tested on other instances of the same iris. CASIA database [47] contains iris images with multiple instances for each iris image. In Figure 34, overlapping points are shown after decoding the vault of instance 1 and comparing it with the other iris instances. The overlapping points are highlighted by green circles. For instance 1, the number of overlapping points is 16 because it is the same template used for encoding. On the other hand, the numbers of overlapping points for the other instances are 3, 6, 4, 6, and 6 respectively. It should be noticed that the total number of points is 16. The overlapping points with the other instances are always less than half of the total points. This is because there is no biometric registration in this algorithm. Biometric

registration makes alignment between biometric image instances to be approximately the same.

In Figure 35, fuzzy-vault is applied for a right eye's iris and is compared to the corresponding left one. The Figure shows that 7 points are overlapping (highlighted by green) and is still lower than half of the total points.

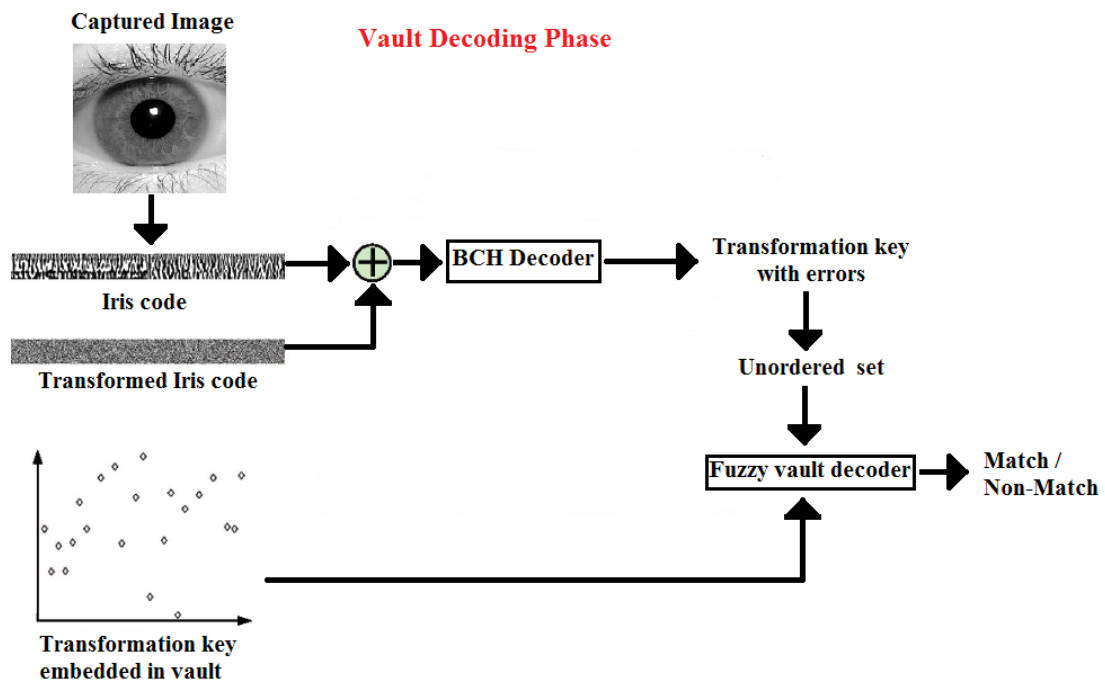


Figure 33: Vault decoding phase.

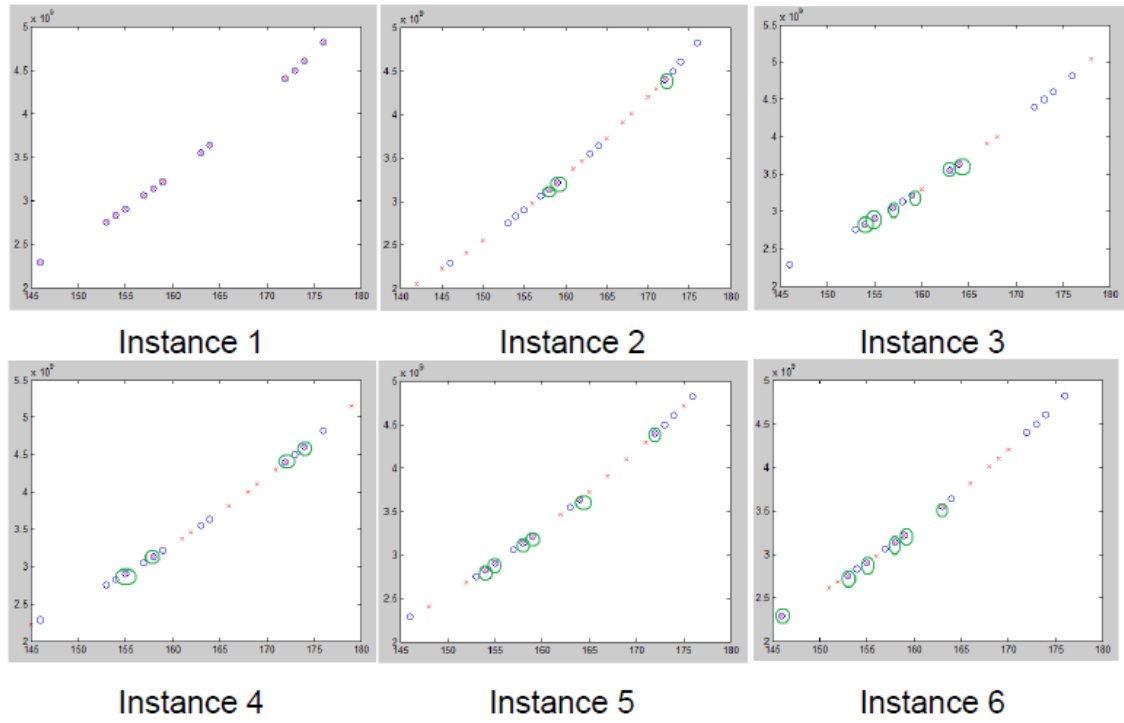


Figure 34: Fuzzy-vault for multiple instances of one iris image.

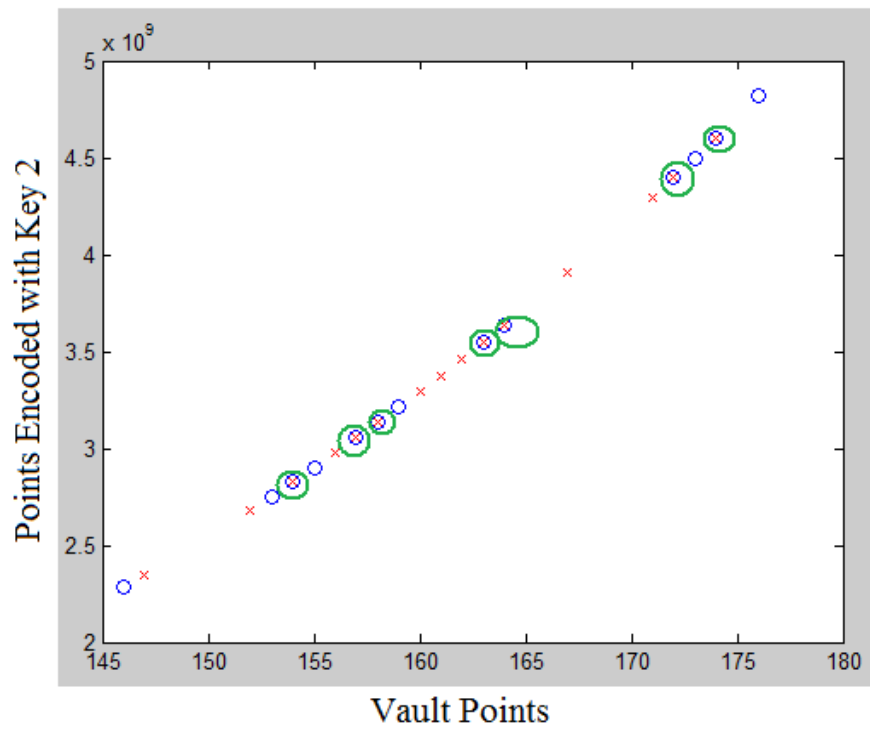


Figure 35: Fuzzy-vault for right and left irises.

4.2 Fingerprint Data Watermarking

In this algorithm, fingerprint data is watermarked using edge process model that has been discussed in detail in Section 3.3. Watermark payload will be hidden in the most salient areas in fingerprint images like curvatures and edges. Before watermarking, many steps need to be carried out. These steps have been summarized as follows:

- 1- Enhance the fingerprint image.
- 2- Detect the core point or the singular point.
- 3- Separate the unused parts from the used ones.
- 4- Apply edge process model to determine edges.
- 5- Apply watermark hiding.

Some fingerprint images have bad quality. This is possibly, due to the sensor surface, or bad quality sensor. For this purpose, the method in [48] has been used for fingerprint image enhancement. This method is based on Short Time Fourier Transform (STFT) analysis. Figure 36 illustrates

an example of bad quality fingerprint image and after its enhancement using STFT analysis.

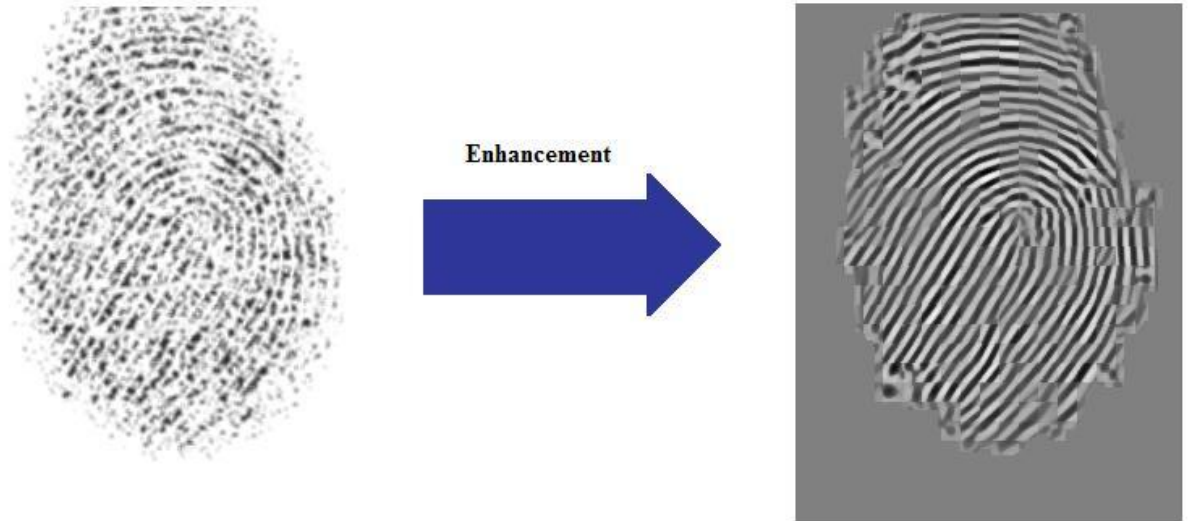


Figure 36: Example of bad quality fingerprint image (left side) and its enhancement using STFT analysis (right side).

In the second step, the singular point or the core point needs to be determined. Singular point region is the area where the ridge curvature is higher than normal and where the direction of the ridge changes rapidly. Poincare index algorithm in [49] has been used for this purpose. This step helps to determine which area is the most important region that should be included in the watermarked image. Figure 37 shows the singular point of fingerprint image in Figure 36.



Figure 37: Singular point detection using Poincare index.

In step three, the unused regions of the fingerprint image need to be separated from the used one. SUSAN algorithm [50] for edge detection has been used for this purpose. This algorithm filters the image based on a predefined threshold. Figure 38 illustrates the fingerprint image in Figure 37 after removing extra parts based on SUSAN algorithm and locating the singular point.

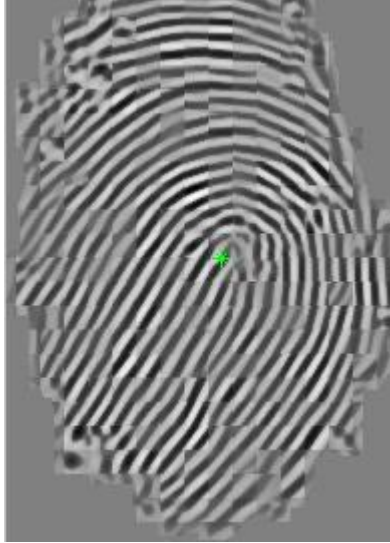


Figure 38: Fingerprint image after removing the extra parts using SUSAN algorithm.

After that, the edge process model is applied to the fingerprint image. The first step in edge process model is to convert the image to the Discrete Wavelet Transform (DWT) domain. Daubechies family of Wavelet Transform has been used. The image should be resized to be a square image before Wavelet Transform. In this case, the image has been resized to be 512×512 . Figure 39 shows the image in Figure 38 after applying resizing and the DWT.

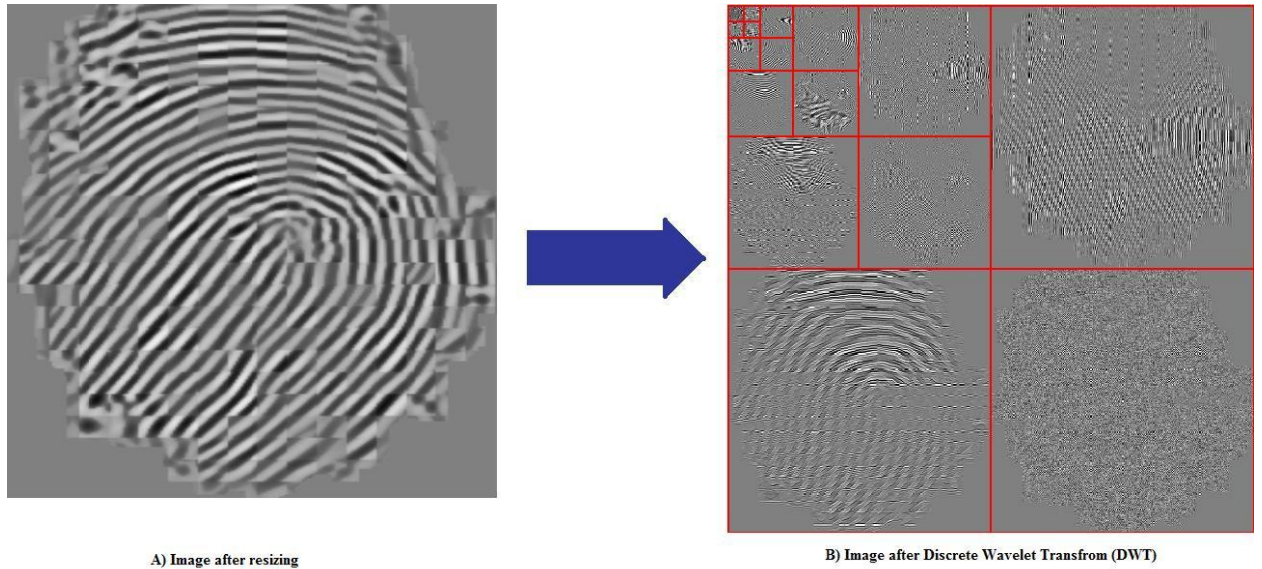


Figure 39: A) Image after resizing. B) Image after applying DWT.

Next, the edge process model is applied to the transformed image. As explained in Section 3.3, the image is classified or quantized based on its variances to separate flat regions from edge and texture regions. Variances of the transformed image are estimated for each subband separately. After that a clustering or quantization process is applied to a quantized image with K channels. 256 channels have been used as clustering bins. Next, UTQ is calculated based on the quantized variances to classify the edges and the flat regions. Figure 40 illustrates how edges appear after applying the edge process model.

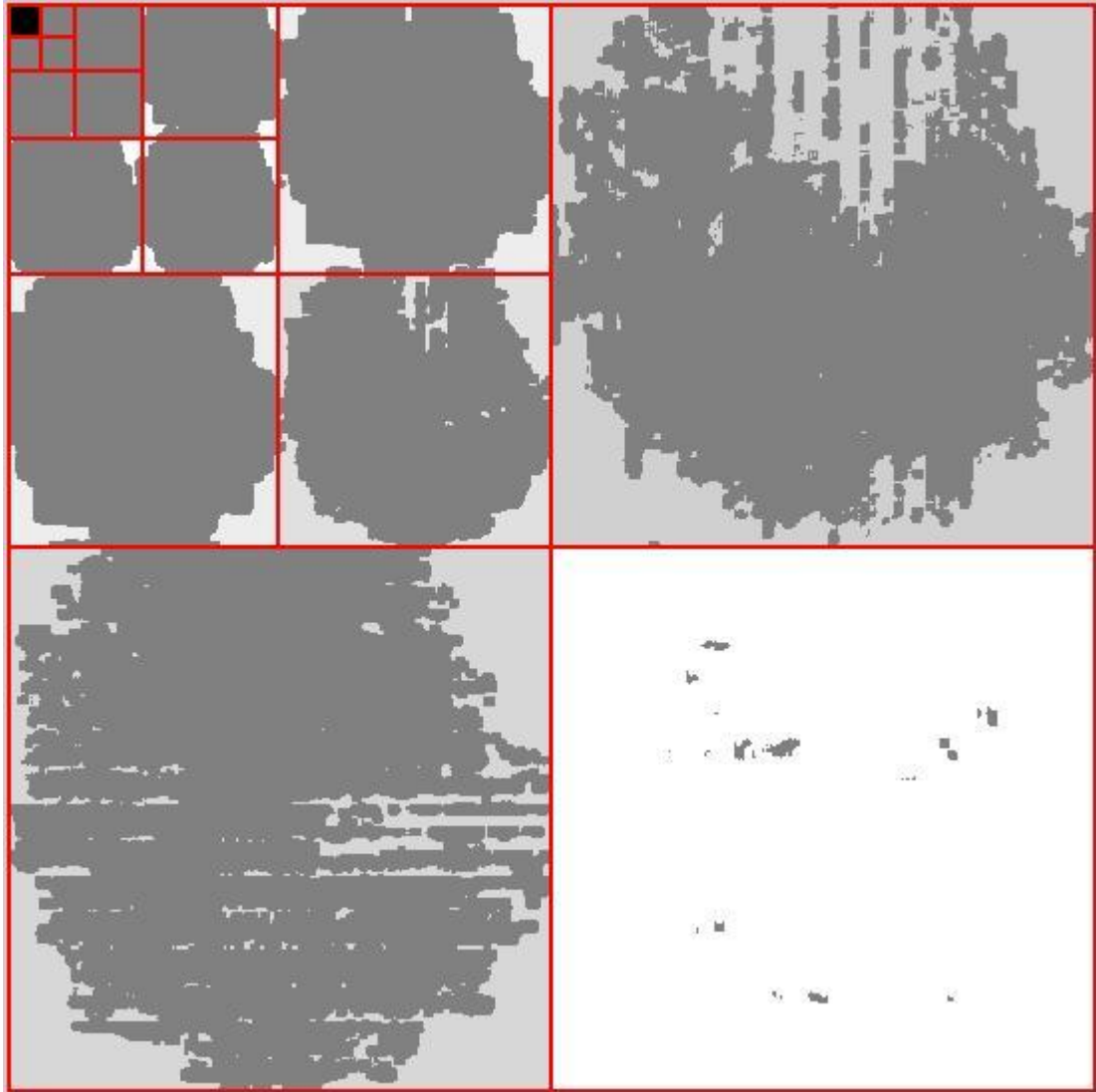


Figure 40: Edges determined using edge process model.

The image is now ready for hiding the watermark into it. The watermark is hidden in the image in the wavelet domain. The used watermark system is illustrated in Figure 41.

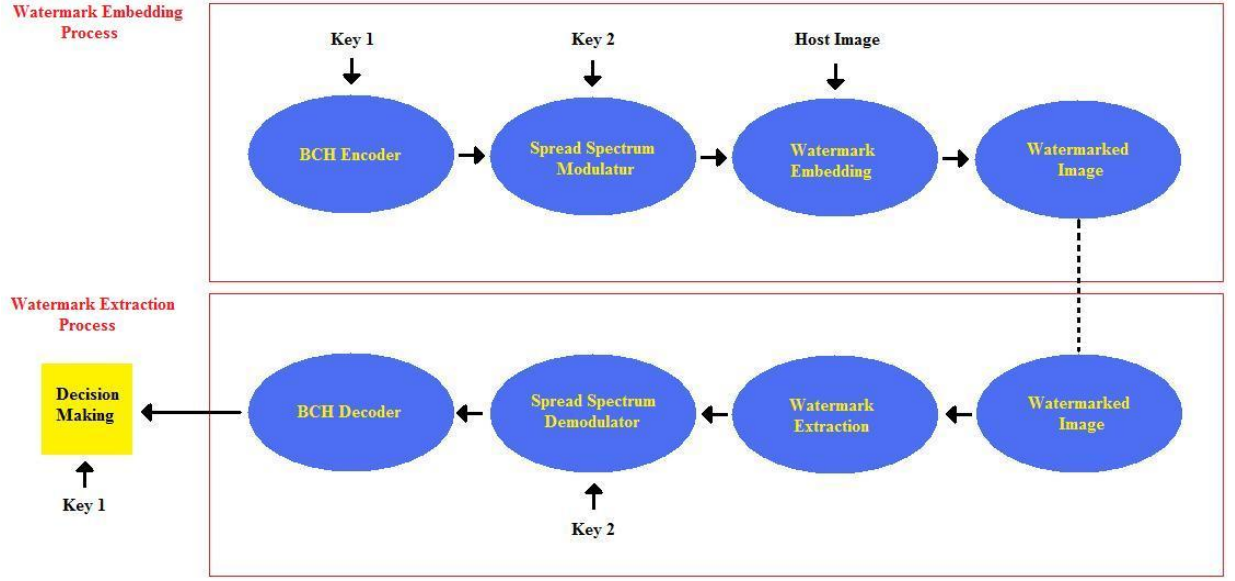


Figure 41: Proposed watermarking system.

As seen from Figure 41, two processes are executed in the watermarking process: watermark embedding and watermark extraction. It is similar to the system proposed by Bastug and Sankur [51], but with BCH encoding being used instead of LDPC encoding.

In the watermark embedding process, two keys are used. Key1 is the payload that is going to be embedded and key2 is used for spread spectrum modulation. Key1 can be an ID, for example, and key2 can be any binary sequence. To increase the watermark robustness, a type of error control code (ECC) is applied to Key1 in binary representation. BCH encoding is used for this purpose. After BCH encoding, data is

modulated with spread spectrum using key2. After that, the watermark is embedded in the host image in the DWT domain. The watermark is going to be hidden into the edges in all subbands except the first subband, the approximation sub-band. Hiding is done according to the following formula:

$$c_i = x_i(1 + \lambda m_i z_i) \quad (27)$$

where i is the i th coefficient, x_i is the original coefficient, λ is the embedding strength, m_i is the ± 1 spread spectrum element of the watermark, and z_i is the watermark value. After watermark embedding, the image needs to be transformed back to the spatial domain. Finally, the image is watermarked.

In the watermark extraction process, the image needs to be transformed to DWT again. Edges have to be synchronized between the sender and the receiver because the extraction is based on the edges. Maximum-Likelihood is used to extract the watermark bits from its footprint coefficients. The DWT channel models the carrier coefficients according to Generalized Gaussian Distribution (GGD). Scale and shape parameters (α_i, β_i) are estimated for each coefficient. An 8×8 window is used for each coefficient to estimate its (α_i, β_i) . Extraction of scale and shape

parameter for GGD is done using ML as described by Minh Do in [39].

After that, the ML decision to extract the watermark bit will be 1 if

$\Delta_{DWT}(r) > 0$ and 0 otherwise. $\Delta_{DWT}(r)$ is given by:

$$\Delta_{DWT}(r) = \ln(1 - \lambda m_i) + \left(\frac{\alpha_i r_i}{1 - \lambda m_i} \right)^{\beta_i} - \ln(1 + \lambda m_i) - \left(\frac{\alpha_i r_i}{1 + \lambda m_i} \right)^{\beta_i} \geq 0 \quad (28)$$

where λ is the embedding strength, and m_i is the ± 1 spread spectrum element for the received coefficient. This process is used for all coefficients that are hiding data in edges.

The previous formula results in a BCH encoded sequence. The watermark is decoded by the BCH decoder to get the key. This key is compared to key1. Then, a decision is made whether it is an authentic fingerprint image or not. Figure 42 shows the original image and the same image after watermarking by using the proposed algorithm. It is noticed from Figure 42 that the images are visually the same and that there is no degradation in quality.



Figure 42: a) Original image b) Watermarked image.

To make sure that the proposed algorithm has not cause a severe degradation for image quality, minutiae points [3] have been identified for the original image, enhanced image, and watermarked image using the proposed algorithm. Figure 43 illustrates the result of minutiae identification. The numbers of minutiae points for the three images are 30, 26 and 27 respectively. The proposed algorithm did not cause a severe degradation as the minutiae points are reduced by 3. Applying this test for multiple images, it is found that watermarked fingerprint images using the edge process model have almost the same number of minutiae points as the enhanced image. Sometimes, the number of minutiae points in the watermarked image is less than that in the original image by 3

points on average. Therefore, the proposed approach does not affect the minutiae points considerably.

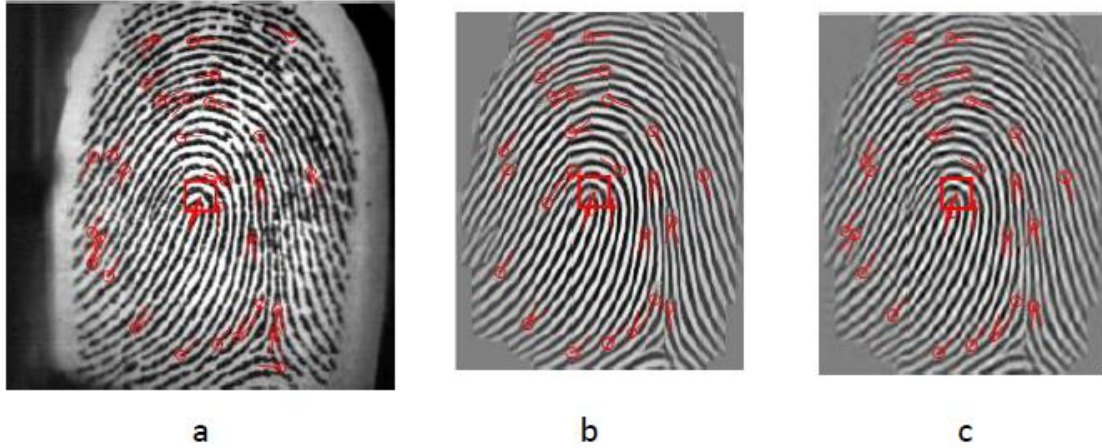


Figure 43: Minutiae points a) Original image b) Enhanced image c) Watermarked image using EP.

4.3 Variations in the Proposed Algorithms

After giving a detailed analysis of the proposed algorithms, it is legitimate to inquire about the possibility of swapping the biometric content used in the proposed techniques. More specifically, a typical question would be: is it possible to embed watermark payload data into iris images and apply fuzzy-vault protection using fingerprint images?

To answer this question, an understanding of the underlying modeling processes is required. First, let's consider Figure 44 which shows the

result of applying the EP modeling on a sample iris image. It is clear that the EP modeling yields very poor results since most of the useful information in a typical iris image consists of textures which cannot be properly modeled by the EP process [44]. On the other side, fingerprint images can be successfully used with the proposed fuzzy vault scheme provided some modifications are carried out to account for novelty.

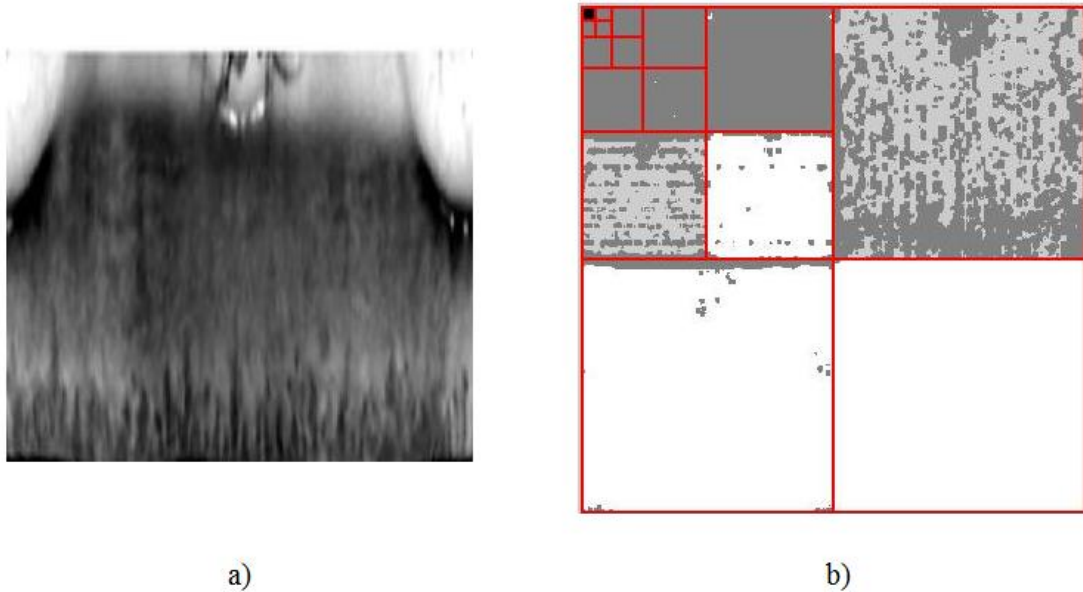


Figure 44: a) sample of normalized iris b) edges map after using the EP model.

CHAPTER 5

EXPERIMENTAL RESULTS

5.1 Introduction

In this Chapter, performance analysis of the proposed algorithms, described in Chapter 4, is carried out in details. First, the proposed fuzzy vault technique for iris template protection is investigated through several experiments. A small iris database is used to conduct these experiments. Cryptanalysis of the proposed method is carried out through several iris image manipulation and alteration techniques. To assess the algorithm robustness, iris images, selected from the iris database, are secured through fuzzy vault. Then, a set of image filters is applied to these iris images. To evaluate the algorithm robustness, codes generated from the fuzzy vault block using unaltered and altered iris image templates are compared using a specific comparison metric.

The assessment of the proposed watermarking algorithm, detailed in Section 4.2, is implemented by embedding the watermark payload in image locations other than those determined by the edge process. The extra embedding is aimed to highlight the robustness of the EP-based watermarking scheme. More specifically, the watermark payload is additively embedded in all wavelet subbands except the approximation subband to achieve watermark imperceptibility. It should be noted that the watermark payload is “selectively” embedded in the strongest wavelet coefficients in the proposed watermarking scheme to meet the conflicting requirements of robustness and imperceptibility. However, it should be clear from the proposed embedding approach reconciles between watermark robustness and imperceptibility at the expense of capacity. Moreover, to “fairly” compare both embedding schemes, different repetition rates are applied to the watermark payload in both cases. Finally, the performance of both schemes is assessed based on watermark decoding after applying a set of image filters similar to that applied in the iris template algorithm. To increase the watermark robustness, error-control codes (ECCs) are applied on the binary format of the payloads before embedding in the coefficients of the wavelet subbands. Therefore, watermark decoding is viewed a data communication problem over noisy

channel where the watermark data represents the signal and the host images represent the noise. It is clear that this is a typical data communication system at very low signal-to-noise ratio (SNR).

The remaining of the Chapter is organized as follows. The biometric iris and fingerprint databases used in the computer experiments are described in Section 5.2. Also, the setups of the computer experiments conducted to assess the performance of the proposed algorithms are described therein. Performance results are reported and analyzed in Section 5.3.

5.2 Experiments and Data Setup

5.2.1 Experiment Setup for Fuzzy-vault-based Iris Template Protection

The fuzzy-vault scheme proposed for the protection of iris templates has been tested using a set of iris images from the CASIA database [47]. Ten test images are used for this purpose. As mentioned in Chapter 4, the iris templates are generated using Daugman's algorithm [11]. Each iris template consists of a binary sequence whose length is 20×480 bits. The variables Z_k and Z_i (see Section 4.1 for details) are set to 16 and 31,

respectively. Consequently, the codes of the iris templates are extended to 20×496 bits. The extended length fits the random key *Key1* (see Section 4.1 for details). **Key1** is partitioned into r components. On each component of the key, an ECC code, BCH (31, 16) code, is applied to increase its robustness.

Template attacks are carried out by the following image filters: blurring, motion, and sharpening filters. Blurring is generated by circular averaging filter of radius ranging from 1 to 10. For the motion filtering, image pixels are increasingly moved from 1 pixel to 10 pixels with no angle effect in place. Similarly, the sharpening filter is gradually applied to assess the robustness of the proposed scheme. It is interesting to note that the applied filters have different effects on the manipulated image. Figure 45 illustrates the effect of the different image filters on the perceived quality of the manipulated image. Image quality is measured using the well-known peak signal to noise ratio (PSNR) measure [52]. Usually, the PSNR measure is given by [52]:

$$PSNR = 10 * \log_{10} \left(\frac{\sum_{i=1}^M \sum_{j=1}^N (I(x, y) - I'(x, y))^2}{M * N} \right) \quad (29)$$

where $I(x, y)$ and $I'(x,y)$ are the $N \times M$ original and manipulated images, respectively. Since, the image difference, given by Equation 26, is an energy measure at the logarithmic scale, the PSNR metric is expressed in Decibels (dB) units.

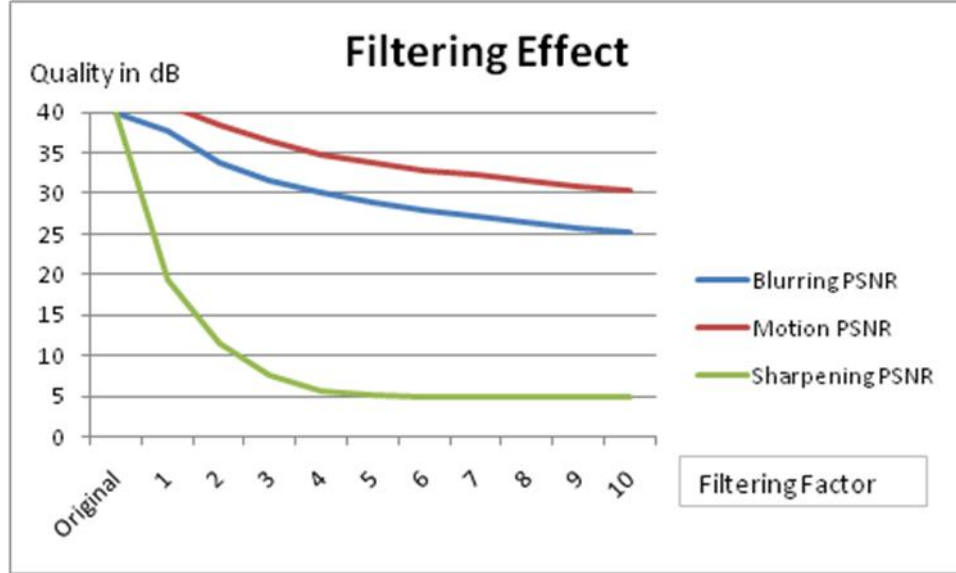


Figure 45: Image filtering effects on perceived image quality.

It is worth noting that a PSNR value of infinity dB is achieved for identical images. Also, a PSNR value below 25 dB would indicate major differences in the perceived quality of the manipulated image with respect to the original one. It is also noted that the blurring and motion filters have similar effects and “slightly” affect the image quality. Unlike blurring and motion filters, the sharpening filter drastically degrades the quality of the manipulated image. Finally, Figure 45 can be used as a

guide to conduct the template attack experiments. More specifically, to mimic “mild” attacks, blurring and motion are preferred while sharpening filters are selected to represent “harsh” attacks. Such scenarios are always considered in cryptanalysis investigations to cover a wide spectrum of attacks ranging from mild to severe.

5.2.2 Experiment Setup for Watermarking-based Fingerprint Image Protection

The performance of watermarking algorithm, described in Section 4.2, is evaluated using a chosen set of fingerprint images. These images are displayed in Figure 46. To restrict the watermark embedding domain to meaningful fingerprint areas, a separation algorithm [50] is used to separate “useful” fingerprint parts from unused ones. The separation technique is threshold-based. Several threshold values have been evaluated. Finally, a threshold value of 10 is selected for its improved performance. After fingerprint segmentation, fingerprint images are transformed using the discrete wavelet transform (DWT). Five (5) decomposition levels are used in conformity with the current practices of DWT-based image coders. At this decomposition level, 16 wavelet

subbands are obtained. To reduce the dynamic range of the subband variances, the latter are quantized to 256 bins yielding 256 watermark embedding channels.

During the watermark embedding process, watermark payloads consisting of 1024 bits are considered. To strengthen the watermark security, BCH (127, 120) codes are used for error control purposes at the encoding stage.

Unlike the filters used in the performance evaluation of the fuzzy-vault-based iris template protection algorithm, more filters are required for the evaluation of the robustness of the proposed watermarking scheme. In fact, seven types of filters are applied on the test fingerprint images. The selected filters are: standard-deviation, median, motion, Wiener, rotation, blurring, and JPEG compression filtering. It is interesting to note that the selected filters not only extend the spectrum of potential attacks but also include a set of unintentional alterations committed by “legitimate” users. For instance, it is a common practice to convert the image format and apply lossy compression provided by most image processing/editing tools widely available. Moreover, JPEG compression attack is viewed as a “combined” lowpass and highpass filtering where different emphasis

weights are used. Finally, to have an “equal-foot” comparison, watermark payloads are embedded using all wavelet subbands and subband edges determined by the edge process. Similar to the experiment, mentioned previously, to quantify the filtering effects on the perceived image quality, the selected filters, defined above, are applied on several fingerprint images and their perceived qualities are measured using the PSNR metric. Figures 47 and 48 illustrate the effects of the seven filters. PSNRs based on image filters using windows of varying lengths are reported in Figure 47. Results using non-window filters are shown in Figure 48.



Figure 46: Set of fingerprint images used.

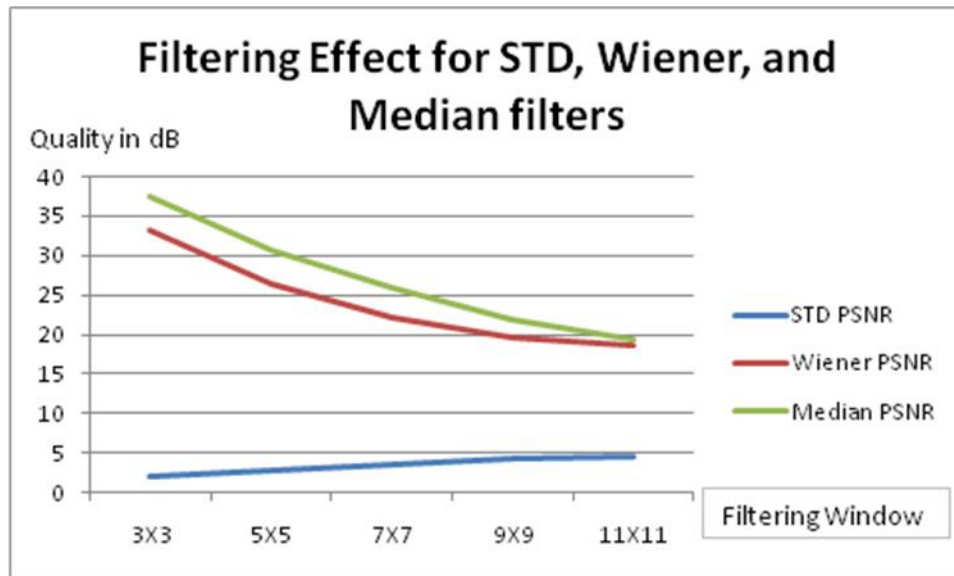


Figure 47: Filtering effects of window-based filters.

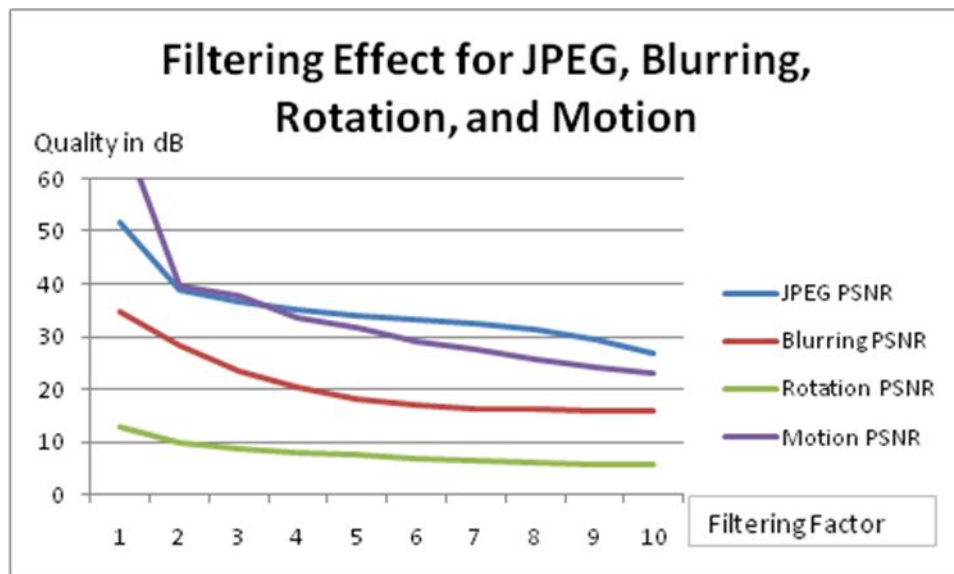


Figure 48: Filtering effects of non-window filters.

It is interesting to notice in Figure 47 that the standard deviation filtering (STD) has severely corrupted the test image unlike Wiener and median filters. Basically, the STD filter replaces every pixel in the manipulated

image by the standard deviation of a block of neighboring pixels. Such alteration results in large-scale pixel modification. However, Wiener and median filters replace each pixel by the average and median of a block of neighboring pixels, respectively. Such alterations bring small-scale pixel modification. Similarly, JPEG filtering has mild effects similar to those caused by Wiener and median filtering. This is due to the fact that JPEG compression can be viewed as a lowpass filtering as mentioned above. On the other hand, rotation filtering has severe effects on the quality of the altered image as indicated in Figure 48. It should be noted that image rotation represents one of the hardest geometric attacks that most of the existing watermarking schemes fail to withstand [32]. Finally, blurring and motion filters are considered as “mild” attacks given the reasonable PSNR yielded by these filters.

Although the attacks assessed in this thesis represent the core of possible intentional and unintentional attacks, the performance of the proposed watermarking scheme has been evaluated and benchmarked using StirMark benchmarking tool [53]. The proposed algorithm achieved good performance. Under mild attacks, the PSNR for attacked watermarked

images achieved around 30 dB. For severe attacks, the PSNR dropped to values ranging around 20 dB.

5.3 Performance Results

5.3.1 Fuzzy-vault-based Iris Template Protection

As indicated in Section 5.2.1, a test sample of ten iris images is used where three types of filtering are applied. Figure 49 illustrates the effect of blurring on a sample iris test image. Blurring is applied using various scales.

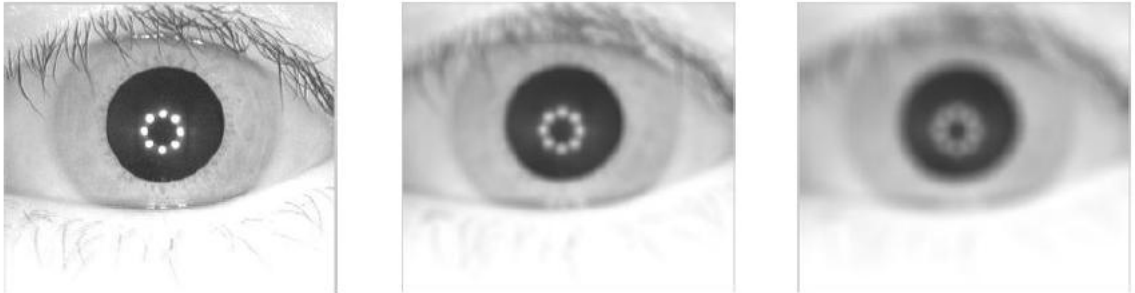


Figure 49: Blurring effects on sample iris image.

To replicate severe adverse environments, the blurring filter has been applied at ten different scales ranging from 1 to 10. Once, the iris code is estimated from the blurred image (attack sample), the number of

overlapping points (see Figure 8 for details) is estimated. Then, the extracted key and the stored vault values are compared for matching purposes. This process is carried out for the entire iris test images along with their blurred version. Figure 52 shows the averaged results for the performance evaluation.

In the case of attack-free conditions, the number of overlapping points is 16 as indicated in Figure 52. It is interesting to note that the number of overlapping points is decreased due to the blurring filtering. More specifically, it decreases to 7 due a blurring filter with angle $\theta = 1$. For higher blurring degrees, the number of overlapping points vanishes completely. Therefore, it is concluded that blurring filtering constitutes a severe attack that drastically degrades the security of the fuzzy-vault-based iris template security. It has been noticed that using less than half of the original overlapping points leads to a meaningless protection effort.

The effect of motion filtering on iris test images is illustrated in Figure 50. Similar to the blurring effect, motion filtering is applied using ten different scales ranging from 1 to 10. According to the PSNR measures, motion filtering does not always compromise the perceived quality of the

iris test images. However, it should be noted, as indicated by Figure 50, that the iris texture component is affected by the motion filtering which would impact the quality of the resulting iris code (iris feature). To illustrate the effects of motion filtering on iris images, Figure 50 shows a sample iris image undergoing the motion filtering process using different scales.

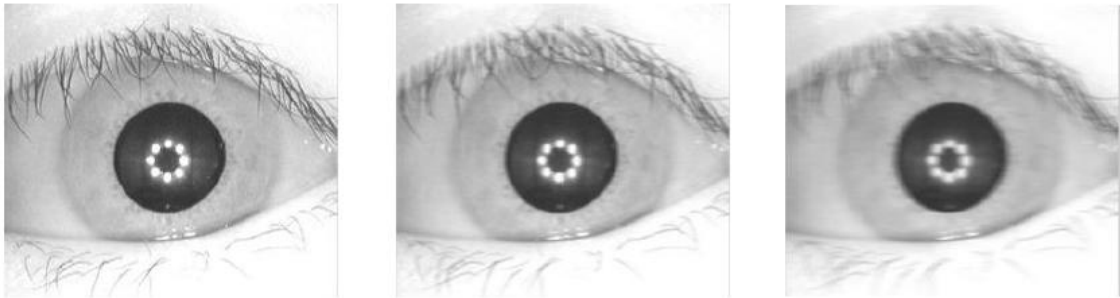


Figure 50: Motion effects on iris images.

Using a motion factor of 1, the quality of iris texture and therefore the iris template is not altered by the motion filtering. In this case, after the extraction of the iris code, the number of overlapping points is 16. This clearly indicates applying mild motions on iris images would not change the iris templates which would imply that both iris images (original and attacked) are declared identical. However, while the more motion is applied on the iris image, the less the number of overlapping points. For instance, Figure 52 shows that for motion factors of 3 and more, the

number of overlapping points would drop to 7 which would mean that the original and attacked iris images will be declared different. The variations in number of overlapping points between original and attacked iris images are reported in Figure 52 for the filtering attacks considered in this performance evaluation. In conclusion, it safe to state that motion filtering has less impact on the performance of the proposed fuzzy-vault-based for the protection of the iris templates than the blurring filtering although the number of overlapping points is dropping by half for motion factors more than 3.

Unlike motion and blurring filtering, sharpening has affected more severely the perceived quality of the iris images as illustrated in Figure 45 by the PSNR measure. Sharpening is applied cumulatively up to ten times on sample iris images. Figure 51 shows the effect of sharpening on the quality of a sample iris image. Using a cumulative factor of 1, the attacked iris image has preserved some of the original iris texture details. However, using a cumulative factor of 5, the texture details have been completely corrupted. In this case, the resulting iris feature (iris codes) would be completely different than that pertaining to the original iris image.

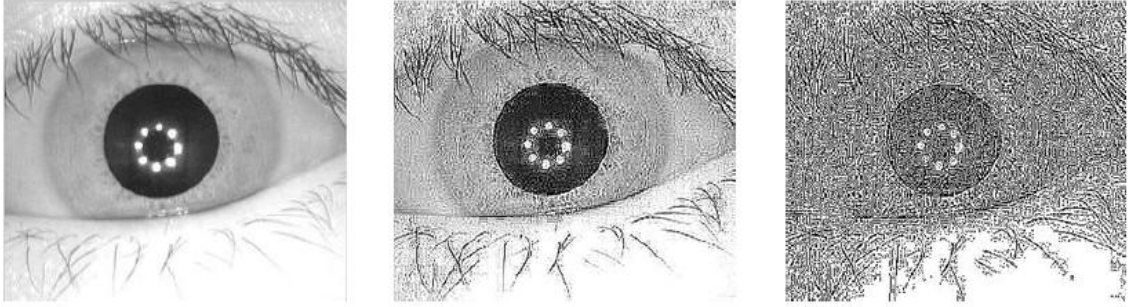


Figure 51: Sharpening effects on iris images.

As expected from Figures 45 and 49, the performance of the proposed fuzzy-vault-based iris template protection algorithm in the presence of sharpening filtering attack is very poor. This finding is clearly indicated by Figure 52. Moreover, unlike the previous filtering process (motion and blurring), sharpening filtering not only deteriorates the perceived quality of the iris images but also drastically reduces the number of overlapping points even at moderate cumulative factors of sharpening.

In summary, fuzzy-vault-based protection is recommended for securing iris templates in the presence of moderate attacks using motion and blurring filtering. However, the proposed scheme performs very poorly in the presence of sharpening filtering attacks as suggested by the findings of the computer experiments carried out in this Chapter.

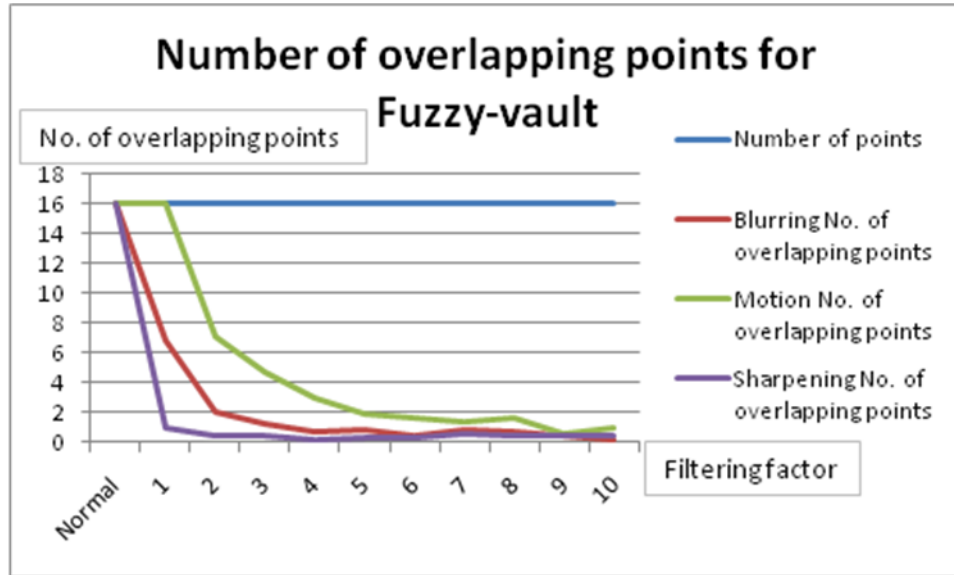


Figure 52: Performance of fuzzy-vault-based iris template protection under various filtering attacks.

5.3.2 Watermarking-based Fingerprint Image Protection

In order to provide a benchmarking framework for the proposed watermarking-based fingerprint protection, watermark embedding is performed using: 1) the coefficients of the entire high-frequency wavelet subbands and; 2) the wavelets coefficients selected by the edge process. It should be noted in both cases, the low-frequency (approximation) subband is not considered for watermark embedding process to ensure watermark imperceptibility.

As mentioned previously in Section 5.2.2, seven types of filters have been applied to the watermarked fingerprint images using different scales and factors. The reported performance results have been averaged over all the fingerprint test images considered in this experiment. To measure the performance of the proposed watermarking scheme, the total number of bits in difference between the original and decoded (recovered) watermark sequences is considered. This measure is known as the bit-error-rate (BER) commonly used in data communication problems. This is reminiscent of a pure digital communication system.

These seven filters have been categorized into three categories where each category is different in its measures and scales. The first category represents the windowed filters, compression filters (JPEG and other) are represented by the second category. The last category represents non-windowed filters.

The first category of windowed filters included three the following filters: 1) standard-deviation filtering (STD); 2) Wiener filtering; and 3) median filtering. In the category of filters, five window sizes have been used ranging from 3x3 to 11x11. The quality of the test fingerprint is most severely damaged by STD filtering as indicated by Figure 53. At

moderate window sizes, STD filtering is viewed as a mild sharpening process. However, using large windows (9×9 and 11×11), the salient points of the fingerprint image (mainly ridges containing minutiae points) are completely altered. In fact, image edges have been fused which would be a challenging task even for robust edge-directed schemes such as the one proposed in this thesis. On the other side, Wiener filtering has little effect on the perceived quality of the test fingerprint image if not improving its quality by removing parts of the background noise which is the very purpose of Wiener filtering. Finally, median filtering acts like a high-pass filtering process (kind of sharpening) where each pixel is replaced by the median value of some neighboring pixels. Usually, this process is favored for mitigating the effects of outliers (edges in the case of images).

Figure 54 summarizes the performance of the decoding process of the watermark payload from watermarked fingerprint images in the presence of STD filtering attack. As hinted by Figure 53, neither the proposed watermarking nor the benchmark algorithm has achieved an acceptable performance. This is mainly due to the fact that most of the fingerprint features are destroyed using this type of attack using moderate and large

window sizes. Under median and Wiener filtering attacks, the proposed watermarking technique has yielded better performance than the benchmark watermarking technique as indicated in Figures 55 and 56.

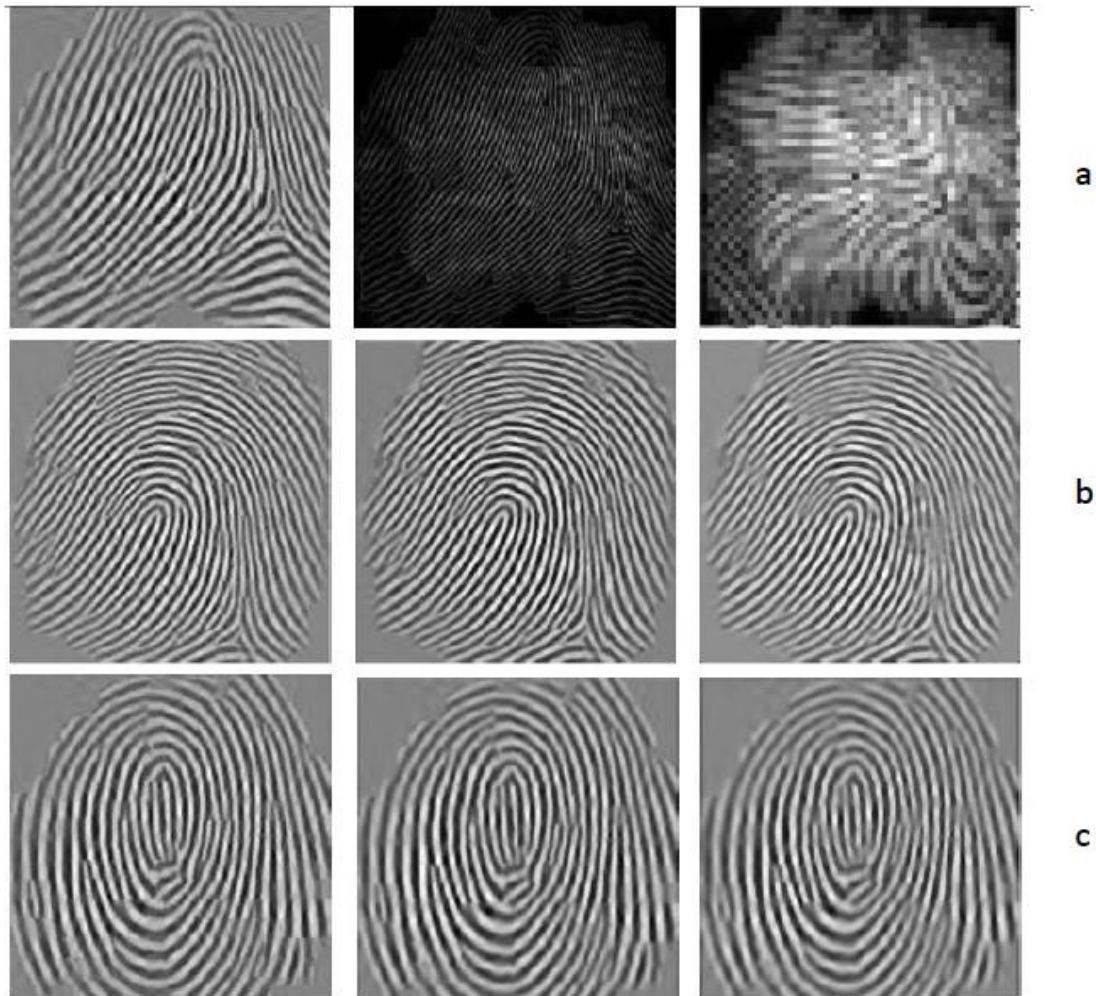


Figure 53: Effects of windowed filters on a sample fingerprint image. (a) STD filtering. (b) Median filtering. (c) Wiener filtering.

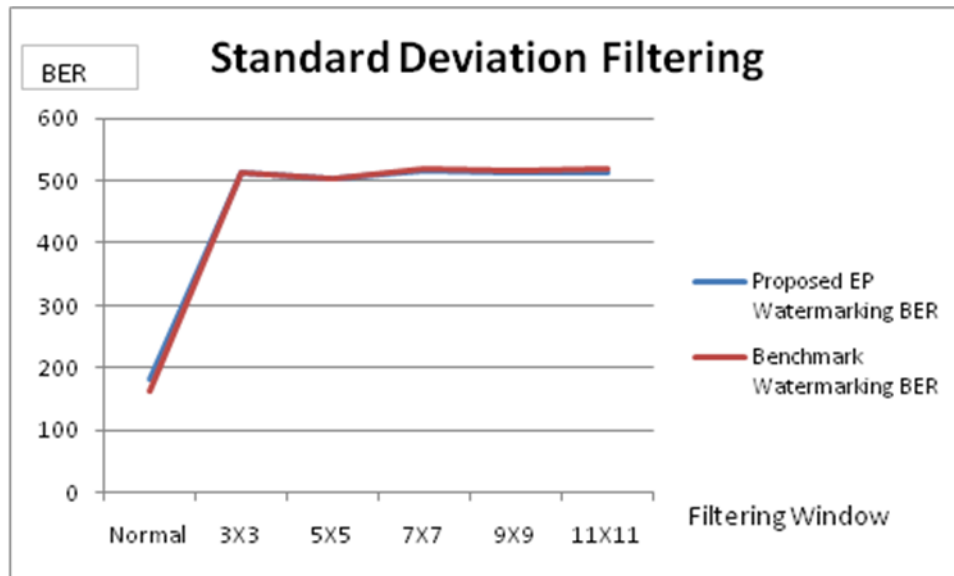


Figure 54: Watermark decoding performance in the presence of STD filtering attack.

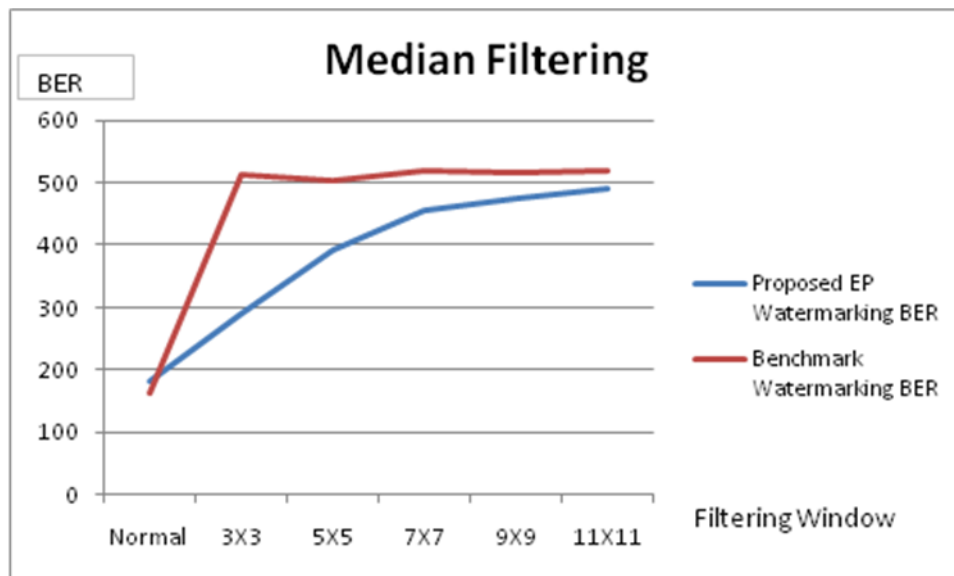


Figure 55: Watermark decoding performance in the presence of median filtering attack.

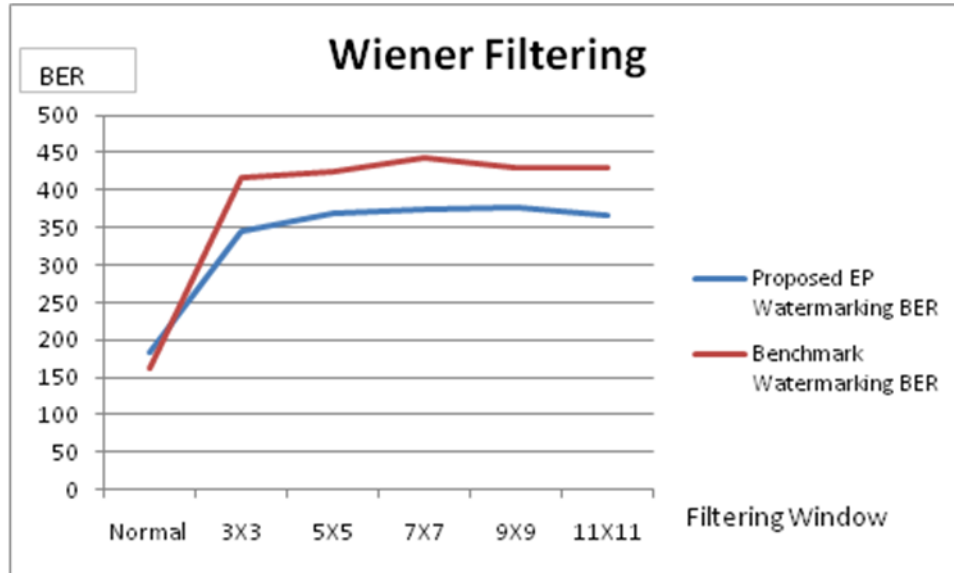


Figure 56: Watermark decoding performance in the presence of Wiener filtering attack.

To assess the performance of the proposed watermarking technique under JPEG compression attack, the watermarked fingerprint images have been compressed using various JPEG quality factors ranging from 100 to 10. A JPEG quality of 100 means a high quality compressed image at a very low compression rate and a value of 1 would mean a highly compressed image at a high compression rate and a poor perceptual quality [52]. Figure 57 illustrates the visual effect of JPEG compression on a watermarked fingerprint image using JPEG quality factors of 100 and 20. Figure 57.a shows the original fingerprint image and the compressed JPEG images at quality factors 100 and 20 are shown in Figure 57.b. and

Figure 57.c, respectively. At such quality factors, the JPEG compression filter has mild and moderate effects on the test fingerprint images.

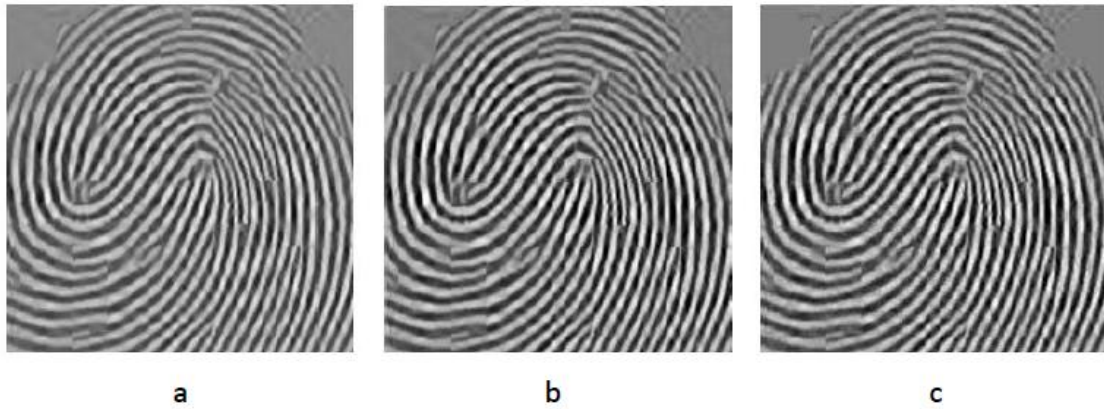


Figure 57: JPEG compression effect. (a) Original image. (b) JPEG compressed image using quality factor of 100. (c) JPEG compressed image using quality factor of 20.

It should be noted from Figure 57 that JPEG compression using such quality factors contributes to the refinement of the fingerprint ridges and contours.

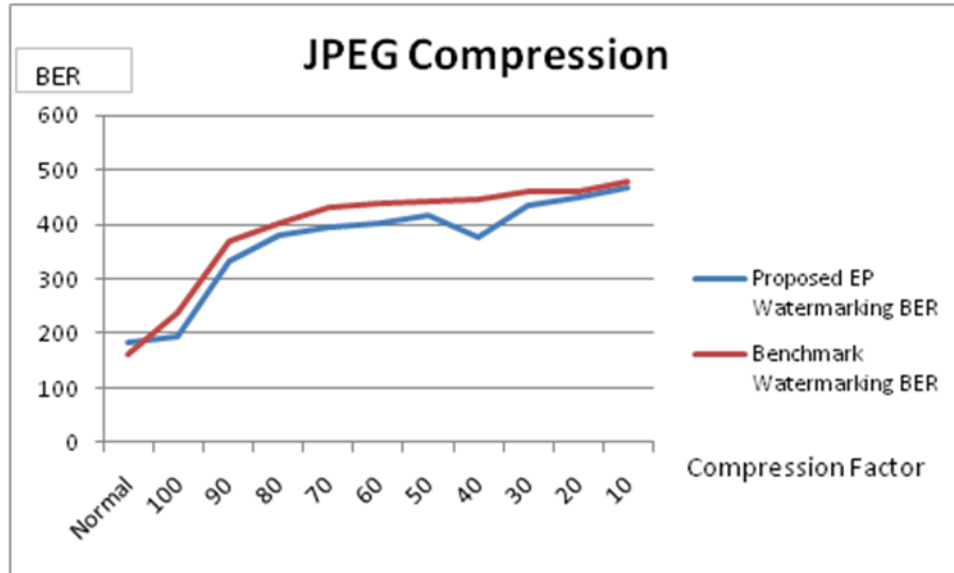


Figure 58: Watermark decoding performance in the presence of JPEG compression attack.

Figure 58 summarizes the performance of the proposed watermarking scheme in the presence of JPEG compression attack. The JPEG quality factor has been varied from 100 (mild attack) to 10 (severe attack). It is clear that the proposed watermarking technique has yielded better performance than the benchmark watermarking technique.

The effects of the filters pertaining to the third category of attacks (blurring, rotation and motion) on a watermarked fingerprint image are illustrated in Figure 59. These filters have been applied on test images using factors ranging from 1 (mild attack) to 10 (severe attack).

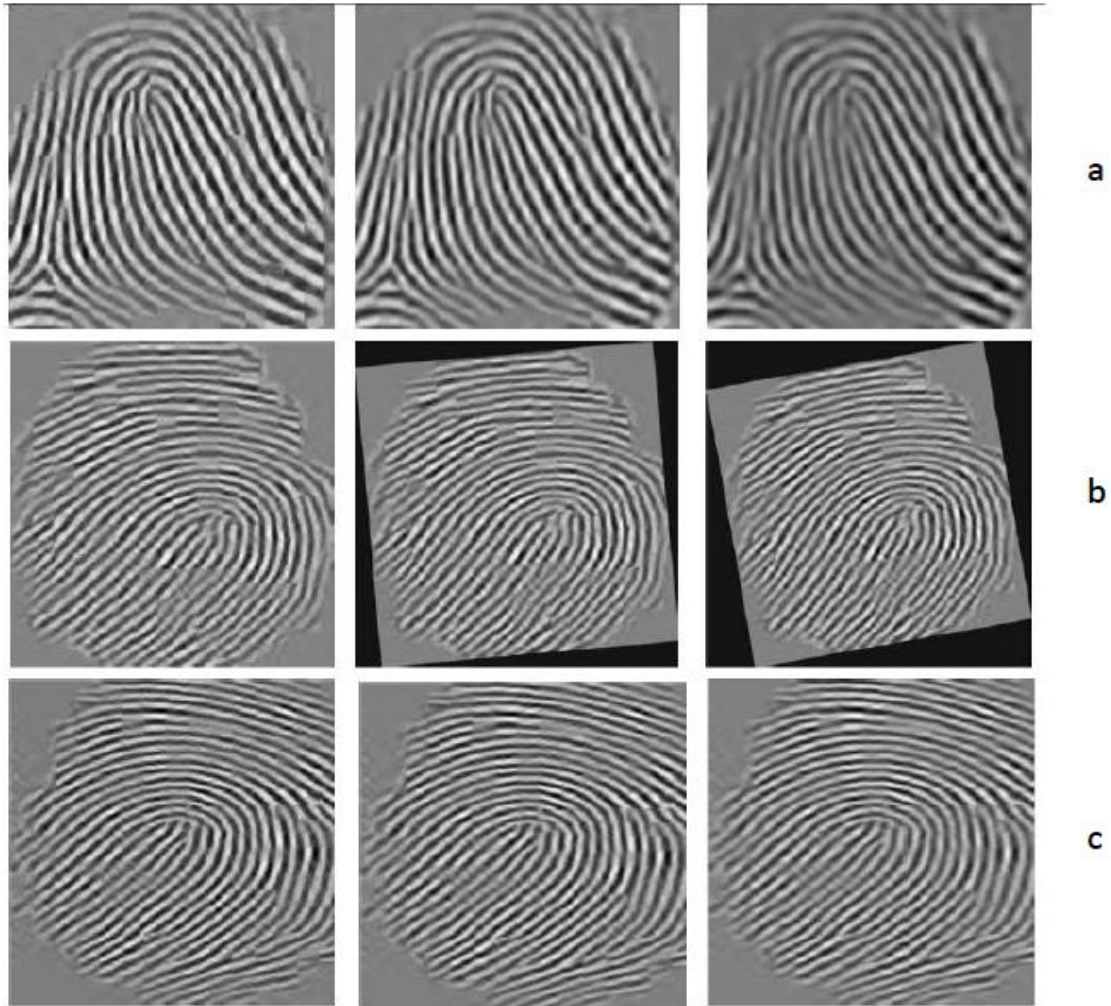


Figure 59: The effect of the third category filters on a watermarked fingerprint image. (a) Effect of blurring filtering attack. (b) Effect of rotation filtering attack. (c) Effect of motion filtering attack.

As indicated by Figure 59, the blurring filtering attack is the most severe attack applied on test fingerprint images in terms of PSNR measures. Moreover, it should be noted that rotation, a well-known geometric attack, is the hardest attack that most of the existing watermarking algorithms easily fail to withstand. Applying motion using simple motion

factors does not affect much the perceptual quality of the watermarked images and therefore has little affect on the performance of the proposed watermarking algorithm as suggested by Figure 62. However, unlike motion filtering attack, blurring and rotation attacks have more negative impacts on the performance of the proposed algorithm as shown in Figures 60 and 61.

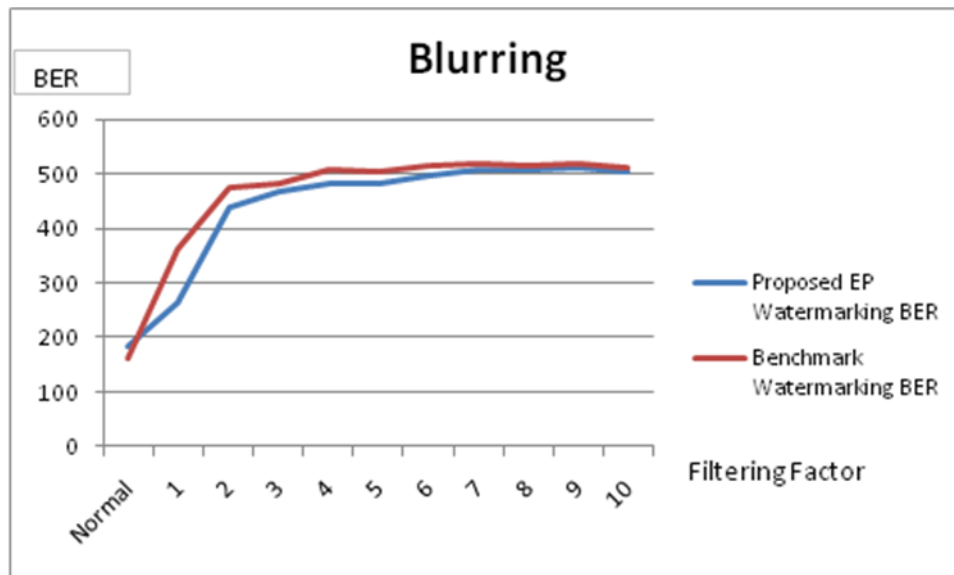


Figure 60: Watermark decoding performance in the presence of blurring filtering attack.

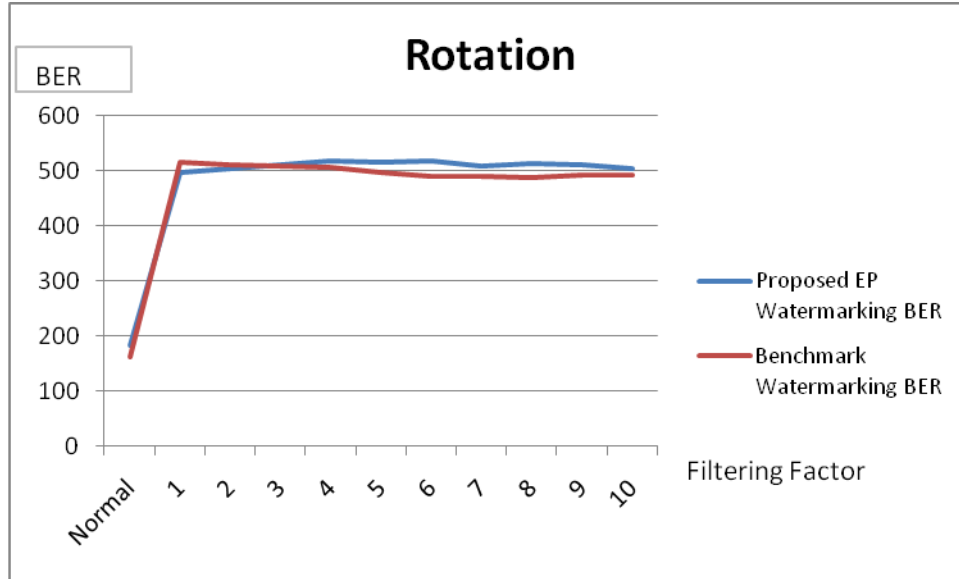


Figure 61: Watermark decoding performance in the presence of rotation filtering attack.

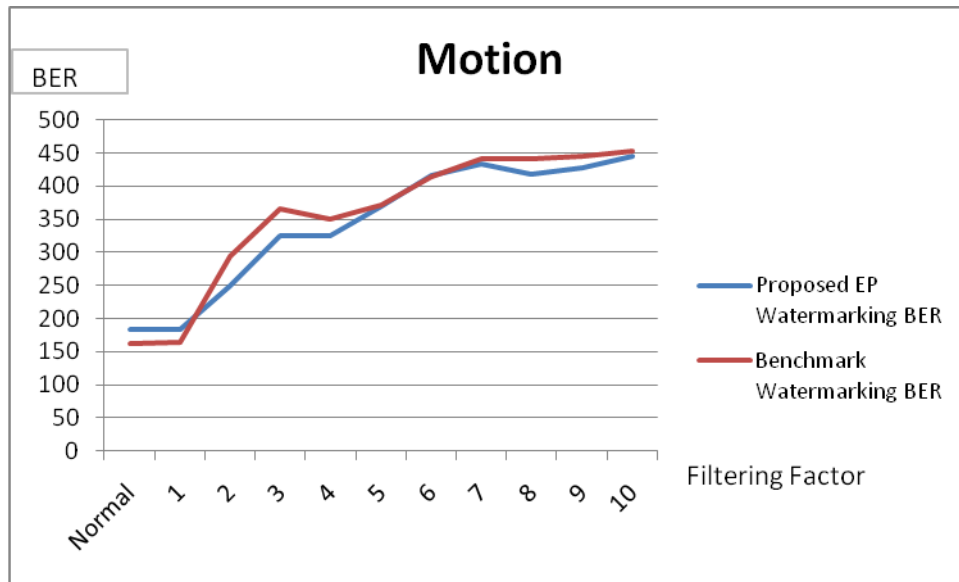


Figure 62: Watermark decoding performance in the presence of motion filtering attack.

In conclusion, based on the results reported in Figures 60-62, the proposed watermarking algorithm has not only yielded better performance than the benchmark watermarking algorithm but it should be

kept in mind that the proposed algorithm embeds the watermark payload in less locations than the benchmark one which would make it more vulnerable to a variety of attacks. However, the reported results clearly indicate that this is not the case. The improvement in performance is mainly contributed to the “selectivity” of the watermark embedding process adopted by the proposed algorithm. The selected subband wavelet coefficients effectively represent the “genuine” robust features of the images being watermarked, i.e., fingerprint edges.

5.3.3 Performance Analysis of EyeCerts®

Algorithm

The functionality of EyeCerts® algorithm, outlined in Section 2.3.1, is implemented to assess the performance of this certification algorithm. Figure 63 gives the details of the implementation diagram of EyeCerts® algorithm adopted in this thesis. It was tested on a set of iris images from CASIA database [47]. It contains seven iris sample images for each subject taken over different sessions where 108 subjects (different people) were enrolled. EyeCerts® [12] makes a binary decision about the

subject authenticity. In this case, two types of errors can be used to assess its performance. These errors are: 1) false positive and 2) false negative alarms. False positive alarm occurs when two iris images pertaining to different subjects are declared otherwise. On the other hand, false negative alarm takes place when two iris images from a same subject are declared belonging to two different subjects. In this performance evaluation, EyeCerts® achieved a false negative rate around 0.05 and a false positive rate of 10^{-5} .

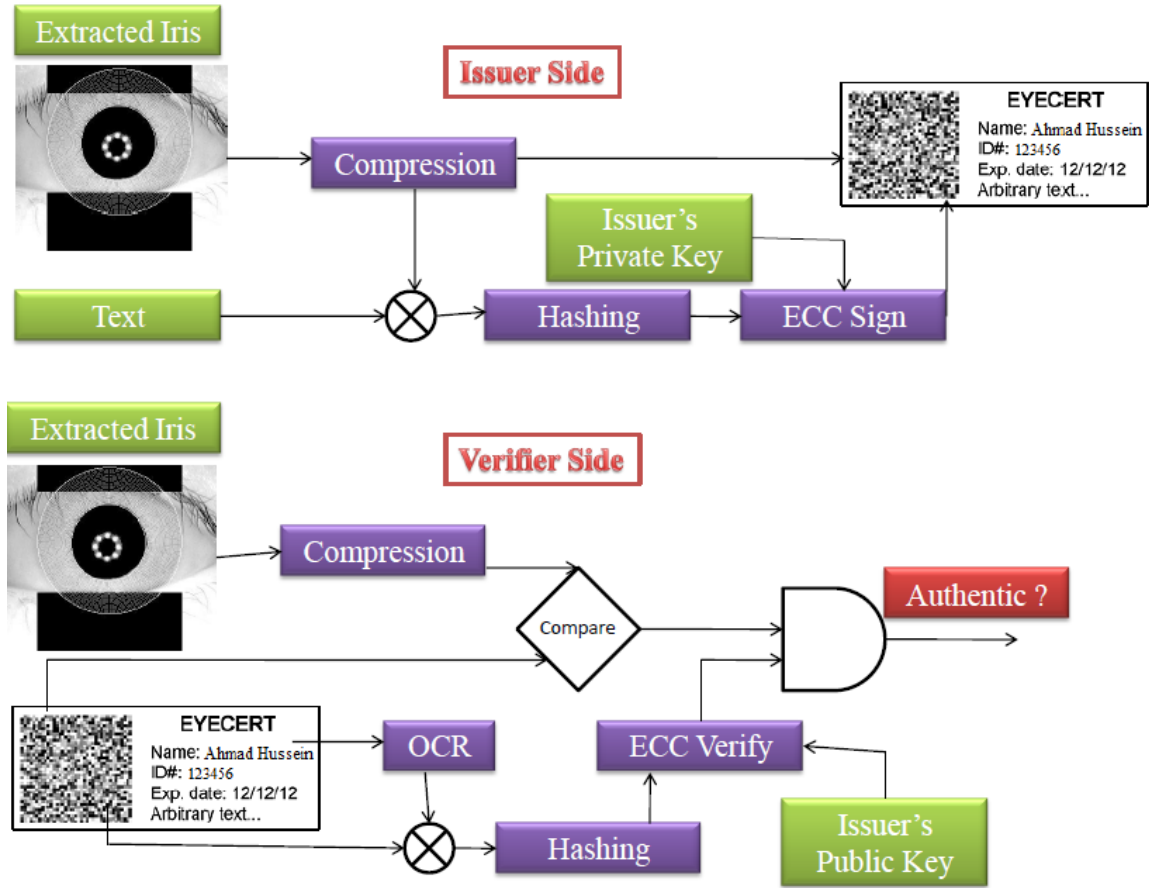


Figure 63: Implementation diagram of EyeCerts® algorithm.

5.3.4 Performance Analysis of FaceCerts®

Algorithm

The functionality of FaceCerts® algorithm [13], outlined in Section 2.3.2, is implemented to assess the performance of this certification algorithm. Figure 64 gives the details of the implementation diagram of FaceCerts® algorithm adopted in this thesis. To circumvent the problems associated

with the face-based recognition algorithms, FaceCerts® does not depend on the face matching results; it relies only on the Euclidean distance between the encoded face image in the barcode and the photo. A performance evaluation has been carried out over 1000+ FaceCerts® demo tests. A false negative rate of less than 10^{-4} has been achieved. Using a different set of 20,400 different pairs of facial photos, another performance evaluation has been carried out achieving a false positive rate of less than 10^{-6} .

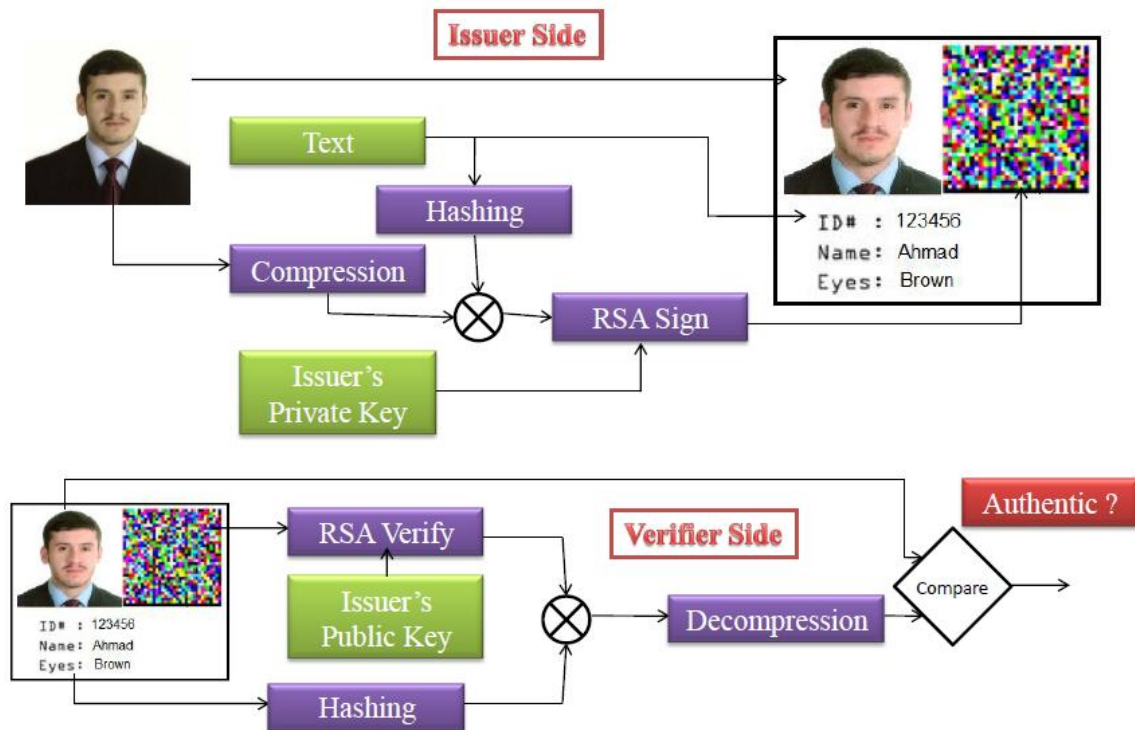


Figure 64: Implementation diagram of FaceCerts® algorithm.

5.4 Conclusions and Summary

In summary to this Chapter, the previous results have shown that fuzzy-vault-based protection is performing well for securing iris templates in the presence of moderate attacks using motion and blurring filtering. On the other hand, it performs very poorly in the presence of sharpening filtering attacks as derived by the findings of the experiments carried out in this Chapter.

The proposed watermarking algorithm has yielded better performance than the benchmark one. Meanwhile, the proposed algorithm embeds the watermark payload in fewer locations which make it more resistible to a variety of attacks in comparison to the benchmark watermarking. However, the reported results clearly indicate that the proposed algorithm has achieved good results while retrieving watermark. On the other hand it is noticed that it very weak against geometrical attacks.

Finally, the two certified biometric systems have been clearly demonstrated and there results have been shown. They achieved a very low rate for false positive and false negative measures.

CHAPTER 6

CONCLUSIONS

This Chapter gives a summary of main thesis contributions and guidelines for possible future work directions based on the presented research findings.

6.1 Summary

Biometric authentication nowadays is becoming a very important task in comparison with traditional authentication techniques. In this thesis, the advantages of biometric authentication, biometric systems and their components, attacks that face biometric systems, and how to secure and evaluate biometric systems have been discussed.

The work of this thesis has been concentrated on securing biometric templates and data. The security schemes for securing biometric data have been demonstrated and compared. These schemes are biometric cryptography, biometric watermarking, and the certification of biometric systems. Fuzzy-vault scheme for securing iris data has been implemented and its robustness has been demonstrated against image filtering attacks. Furthermore, a new technique for fingerprint watermarking has been developed and its performance evaluated. Finally the results of evaluation have been presented.

6.2 Future Work

In the future, this work can be extended to secure other biometric data like face and palm-print, for example. Fuzzy-vault has just been implemented for securing fingerprints and iris templates. In the future, it can be extended to secure eigenfaces. Furthermore, we noticed that the fuzzy-vault scheme for iris templates is very weak against attacks like blurring, motion and sharpening. This can be improved in the future to be more robust to these attacks and other attacks.

For the edge process watermarking, it can be extended to work with more biometric data images. It was noticed also that the edge process watermarking is very weak against geometrical attacks. This work can be improved to tolerate these types of attacks. Furthermore, edges need to be synchronized between the sender and the receiver or between the issuer and the authenticator. This is sometimes will be more boring. So this wants to be improved to let the authenticator extracts the watermark without synchronizing with the issuer. Finally, the Error Correcting Code (ECC) schemes are still weak and cannot achieve a very good precision. Their precision is somehow acceptable but still returning big error rates. This problem should be taken into consideration in the future to propose a new ECC that achieves a good precision for correcting errors.

REFERENCES

- [1] A. K. Jain, P. Flynn, and A. A. Ross. *Handbook of Biometrics*, Springer, 2008.
- [2] Wikipedia, “*Daubert standard*,” December 2008. [online]. Available: http://en.wikipedia.org/wiki/Daubert_standard
- [3] N. K. Ratha, J. H. Connell, and R. M. Bolle, “*An Analysis of minutiae matching strength*,” *Lecture Notes in Computer Science*, vol. 2091, pp. 223-228, 2001.
- [4] A. Juels and M. Sudan, “*A Fuzzy vault scheme*,” *IEEE International Symposium on Information Theory*, pp. 408-425, 2002.
- [5] A. K. Jain, K. Nandakumar, and A. Nagar, “*Biometric template security*,” *Eurasip Journal on Advances in Signal Processing*, Special Issue on Biometrics, January 2008.
- [6] U. Uludag and A. Jain, “*Fingerprint-based fuzzy vault*,” [online]. Available: <http://www.biometrics.org/bc2005/Presentations/Conference/1%20Mond>

[ay%20September%2019/Mon_Ballroom%20A/RS_%20mutUludag_final.pdf](#)

[7] U. Uludag, S. Pankanti, and A. Jain, "*Fuzzy vault for fingerprints*," Springer Berlin / Heidelberg, vol. 3546, pp.310-319, June 2005.

[8] K. Nandakumar, A. Jain, and S. Pankanti, "*Fingerprint-based fuzzy vault*," IEEE Transactions on Information Forensics and Security, vol. 2, pp. 744-757, November 2007.

[9] K. Nandakumar, "*Multi-biometric systems: fusion strategies and template security*," PhD Dissertation, Michigan State University, 2008.

[10] L. Masek, "*Recognition of human iris patterns for biometric identification*," Master Thesis, the University of Western Australia, Australia, 2003.

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

[12] D. Schonberg and D. Kirovski, "*EyeCerts*," IEEE Trans. on Information Forensics and Security, vol. 1, no. 2, pp. 144-153, June 2006.

- [13] D. Kirovski and N. Joji, “*Cryptographically secure identity certificates*,” IEEE International Conference of Acoustics, Speech and Signal Processing (ICASSP), vol. 4, pp. 413-416, 2004.
- [14] M. Turk and A. Pentland, “*Face recognition using eigenfaces*,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 586-591, June 1991.
- [15] J. Shlens, “*A Tutorial on principal component analysis*,” version 3.01, April 2009, [online]. Available:

<http://www.sn1.salk.edu/~shlens/pub/notes/pca.pdf>
- [16] E. F. Hembrooke, “*Identification of sound and like signals*,” United States Patent No. 3004104, 1961.
- [17] G. Langelaar, I. Setyawan and R. Legendijk, “*Watermarking of digital image and video data: A state-of-the-art overview*,” IEEE Signal Processing Magazine, vol. 17, no. 5, pp. 20-46, September 2000.
- [18] I. J. Cox, M. L. Miller, J. A. Bloom, J. Fridrich, and T. Kalker. *Digital Watermarking and Steganography*. Morgan Kaufmann, Second Edition, 2008.

- [19] F. Hartung and M. Kutter, "*Multimedia watermarking techniques*," Proceedings of the IEEE, vol. 87, no. 7, pp. 1079-1107, July 1999.
- [20] D. W. Stouch, "*A Survey of practical applications in image watermarking*," Technical Report, Cambridge and Boston University Metropolitan College, 2006.
- [21] T. Lan, M. Mansour and A. Tewfik, "*Robust high capacity data embedding*," IEEE International Conference on Image Processing (ICIP), vol. 1, pp. 581-584, September 2000.
- [22] B. Chen and C. W. Sundberg, "*Broadcasting data in the FM band by means of adaptive contiguous band insertion and precanceling techniques*," IEEE International Conference on Communications (ICC), vol. 2, pp. 823-827, June 1999.
- [23] H. C. Papadopoulos, A. Haralabos and C. W. Sundberg, "*Simultaneous broadcasting of analog FM and digital audio signals by means of adaptive precanceling techniques*," IEEE Trans. on Communications, vol. 46, no. 9, pp. 1233-1242, September 1998.
- [24] P. Loo, "*Digital watermarking using complex wavelets*," PhD Thesis, University of Cambridge, UK, 2002.

- [25] N. K. Ratha, J. H. Connel and R. M. Bolle, “*Secure data hiding in wavelet compressed fingerprint images*,” Proceedings of the 2000 ACM Workshop on Multimedia, pp. 127-130, 2000.
- [26] S. Pankati and M. M. Yeung, “*Verification watermark on fingerprint recognition and retrieval*,” SPIE, Security and Watermarking of Multimedia Contents, vol. 3657, pp.66-78, April 1999.
- [27] B. Gunsel, U. Uludag and A. M. Tekalp, “*Robust watermarking of fingerprint images*,” Pattern Recognition, vol. 35, no. 12, pp. 2739-2747, December 2002.
- [28] C. Y. Low, A. B. J. Teoh and C. Tee, “*A Preliminary study on biometric watermarking for offline handwritten signature*,” IEEE International Conference on Telecommunications and Malaysia International Conference on Communications, ICT-MICC 2007, pp. 691-696, May 2007.
- [29] C-Y. Low, A. B-J. Teoh and C. Tee, “*Support vector machines (SVM)-based biometric watermarking for offline handwritten signature*,” 3rd IEEE Conference on Industrial Electronics and Applications, ICIEA 2008, pp. 2095-2100, June 2008.

- [30] T. Hoang, D. Tran and D. Sharma, “*Bit priority-based biometric watermarking*,” Second International Conference on Communications and Electronics (ICCE 2008), pp. 191-195, June 2008.
- [31] S. Jung, D. Lee, S. Lee, and J. Paik, “*Fingerprint watermarking for H.264 streaming media*,” Frontiers in the Convergence of Bioscience and Information Technologies, FBIT 2007, pp. 671-675, October 2007.
- [32] K. Zebbiche, L. Ghouti, F. Khelifi, and A. Bouridane, ‘*Protecting fingerprint data using watermarking*,’ Proceedings IEEE of the First NASA/ESA Conference on Adaptive Hardware and Systems (AHS'06), Istanbul, pp. 451-456, June 2006.
- [33] U. Uludag, “*Secure biometric systems*,” PhD Dissertation, Michigan State University, 2006.
- [34] R. Wildes, J. Asmuth, G. Green, S. Hsu, R. Kolczynski, J. Matey, S. McBride. “*A system for automated iris recognition*,” Proceedings IEEE Workshop on Applications of Computer Vision, Sarasota, FL, pp. 121-128, 1994.
- [35] A. Graps, “*An introduction to wavelets*,” IEEE Computational Science & Engineering, vol. 2, pp. 50-61, 1995.

- [36] Ch. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.
- [37] Y. Raja, S. McKenna, and S. Gong, "*Segmentation and tracking using color mixture models*," Asian Conference on Computer Vision, January 1998.
- [38] S. McKenna, Y. Raja, and S. Gong, "*Object tracking using adaptive color mixture models*," Advances in Color Machine Vision, ACCV Spec. Sess., January 1998
- [39] M. Do and M. Vetterli, "*Wavelet-based texture retrieval using generalized Gaussian density and Kullback-Leibler distance*," IEEE Trans on Image Processing, vol. 11, no.2, pp. 946-957, February 2002.
- [40] M. Varanasi and B. Aazhang, "*Parametric generalized Gaussian density estimation*," The Journal of the Acoustical Society of America, vol. 86, pp. 1404-1415, October 1989.
- [41] P. Moulin and M. Mihçak, "*A framework for evaluating the data hiding capacity of image sources*," IEEE Trans. Image Processing, vol. 11, no. 9, pp. 1029-1042, September 2002.

- [42] S. Lo Presto, K. Ramchandran, and M. Orhard, "*Image coding based on mixture modeling of wavelet and a fast estimation-quantization framework*," Data Compression Conf, Snowbird, UT, USA, pp. 221-230, March 1997.
- [43] C. Weidmann and M. Vetterli, "*Rate-distortion analysis of spike processes*," Data Compression Conference, Snowbird, UT, USA, pp. 82-91, March 1999.
- [44] S. Voloshynovskiy, O. Koval, M. K. Mihçak, and T. Pun, "*The Edge process model and its application to information-hiding Capacity Analysis*," IEEE Trans. on Signal Processing, vol. 54, no. 5, pp. 1813-1825, May 2006.
- [45] S. Voloshynovskiy, O. Koval, and T. Pun, "*Wavelet-based image denoising using nonstationary stochastic geometrical image priors*," Image and Video Communications and Processing, vol. 5022, pp. 675-687, May 2003.
- [46] J. Liu and P. Moulin, "*Analysis of interscale and intrascale dependencies between image wavelet coefficients*," IEEE Int. Conf. Image Processing (ICIP), vol. 1, pp. 669-672, October 2000.

[47] CASIA Iris Image Database [Online]. Available:
<http://www.sinobiometrics.com>

[48] S. Chikkerur, V. Govindaraju, and A. Cartwright, "*Fingerprint image enhancement using STFT analysis*," Pattern Recognition, Elsevier Science Inc, New York, USA, vol. 40, pp. 198-211, January 2007.

[49] J. Bo, T. Ping, and X. Lan, "*Fingerprint singular point detection algorithm by Poincare index*," Wenas Trans. on Systems, vol. 7, pp. 1453-1462, December 2008.

[50] S. Smith and J. Brady, "*SUSAN – A new approach to low level image processing*," International Journal of Computer Vision, vol. 23, pp. 45-78, May 1997.

[51] A. Bastug and B. Sankur, "*Improving the payload of watermarking channels via LDPC coding*," IEEE Signal Processing Letters, vol. 11, pp. 90-92, February 2004.

[52] R. C. Gonzalez and R. E. Woods, Digital Image Processing, Prentice Hall, 2001.

[53] F. A. Petitcolas, "*Watermarking schemes evaluation*," IEEE Signal Processing, vol. 17, no. 5, pp. 58–64, September 2000.

VITAE

Name: Ahmad Mahmoud Khalil Hussein

Address: P.O. Box 3530 Amman 11953 Jordan

E-mail: ahmado.hu@gmail.com

Nationality: Jordanian

Education

Sep 2006 – Jan 2010: Master of Science in Computer Science, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia.

Oct 2002 – Feb 2006: Bachelor of Science in Computer Information Systems, University of Jordan, Amman, Jordan.

Professional Experience

Sep 2006 – Oct 2008: Research Assistant, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia.

June 2008 – June 2009: ERP Technical Consultant, Columbus IT, Dammam, Saudi Arabia.