University of Technology, Sydney

Faculty of Engineering and Information Technology

FEIT

Synergising Fingerprint Biometrics and Cryptography for Improved Authentication

By

Nazia Mastali

A thesis submitted in partial fulfilment of the requirements for the degree of Master Degree in the Faculty of Engineering and Information Technology, UTS

Supervisor

Professor Massimo Piccardi

CERTIFICATE OF ORIGINAL AUTHORSHIP

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Friday, 12 April 2013

Dedication

I dedicate this Thesis to my beloved mother.

Mother, words cannot express how grateful I am to you for all the sacrifices that you've made throughout your life, to nurse and nurture me.

I can never make it up to you, not even for a single sleepless night that I have caused you.

Not even for a single one of your advices that are full of wisdom and life lessons which I will always carry with me to help me in life.

Not even for one of your love filled caresses.

I just want to make you proud of the woman that you have raised me to become.

Mother, but no matter what, I will always be your little girl.

Acknowledgement

I would like to express my special appreciation and thanks to my advisor Professor Massimo Piccardi. You have been a tremendous mentor for me. Thank you for encouraging me to pursue this work and for allowing me to grow as a research scientist and thank you for being so kind and supportive when I had no one else to go to. Your advice on both research as well as on my career have been priceless.

I am also very grateful for the support and guidance of two of the best and kindest ever professors and human beings, Professor Bijan Samali and Professor Farzad Safaei. They never turned away from giving me moral and advisory help and guidance, on every occasion that I appealed to them. They always treat every student who, like me, is lucky enough to know them, as their own student and with the uttermost respect and goodwill. Thank you for being such lovely fellows.

The last but not the least, I would like to state my deepest and most sincere gratitude's to my beloved husband, Dr. Bahram Kimiaghalam, who spent sleepless nights and frantic days helping me in my work and was always my greatest, most reliable and most gentle and loving support, especially at the moments when I needed it most. I love him beyond imagination.

Table of Contents

List of Figures

List of Tables:

Abstract

With the advances in Information Technology (IT) there has been an increase in threats to the communication systems and their assets. One of the most important issues of all Internet Protocol (IP) networks that integrate wireless and wired technologies is the applicability and performance of the electronic identification and authentication methods. These schemes employ a variety of technologies of different degrees of security. Cryptography and biometrics are identified as two of the most important aspects of digital security environments.

Biometrics technology nowadays is typically considered a security necessity, tightly coupled with the foundation of highly secure identification and authentication solutions. Also, a biometric system itself is vulnerable to a number of threats. A critical issue in biometric system is to protect the template of a user which is usually stored in a database or a smart card. While cryptography is a powerful tool to accomplish information security, one of the main challenges in crypto systems is to maintain the secrecy of the cryptographic keys.

The fuzzy vault construct is a biometric cryptosystem that secures both the secret key and the biometric template by binding them within a cryptographic framework. The ability to work with the fuzzy data which is common in biometric systems makes this method a promising solution for biometric cryptosystems.

In many applications, fingerprint has been chosen as a core biometric for the fuzzy vault construction. In this thesis, fingerprint has been selected for further study due to its maturity in terms of availability, uniqueness, permanence, feasibility, ease of use and acceptance. It is expected to address some of the limitations in fingerprint fuzzy vault construction by modifying this structure.

Finally, the main contribution of this work is two-fold as follows:

1. An exhaustive review study of the current state of the art in utilising biometrics and cryptography for authentication, more specifically for fingerprint biometric.

2. Proposing a novel method in fingerprints curve extraction which would improve upon current methods on computational load while preserving the required precision. Fingerprints curve extraction is a vital function for both fingerprint classification and extracting the so-called “helper data”. Helper data are required in fuzzy vault implementation.

CHAPTER 1   INTRODUCTION TO THE RESEARCH

In today's electronically wired information era, there are more and more situations which require the identity of an individual, as a user, to be verified by an electronic machine as in the case of transaction authentication or physical or virtual access control (Shukla & Tiwari 2008). Also, the advances in information technology have resulted in increased threats to the communication and information systems and assets and therefore there has been a need to improve the security measures. Furthermore, the need for more and more reliable user authentication techniques has increased concerns about security because of the recent, rapid advancements in networking, communication and mobility (Bala 2008).

Major components of security are Confidentiality, Integrity and Availability (CIA). Confidentiality is defined as the assurance of data privacy, whereby only the intended recipient should be able to hear or view the data. Integrity guarantees that the transmission has not been altered by adding or deleting data from its origin to its intended destination. Last, availability is assertion that IT resources are readily available to authorized users when needed (Boatwright & Luo 2007).

Authentication is the process of determining whether someone is, in fact who they are declared to be. User authentication has mostly been conducted in three ways:

- An authentication can be performed using memorized knowledge, like a pin or password. This is known as "something you know".
- It can be done using some tangible and visible identification like a passport, ID card or driver's license; this is known as "something you have".

- It can also be done comparing a person’s features like fingerprints, signature or voice. This is known as “something you are”, and is also known as Biometric (Bala 2008).

Using individual passwords, ID numbers such as token or pin identification have deficiencies that restrict their applicability in a widely-networked society. The main problem with these numbers and tokens is that they can be stolen and used by unauthorized persons (Bhattacharyya et al. 2009). Furthermore, there exist the difficult task of memorizing numerous passwords and/or PIN numbers in order to access various databases and systems. More often, it becomes almost impossible to remember the different passwords or pass phrases due to the variable constraints on acceptable passwords (minimum/maximum length, minimum number of lower-case letters, minimum number of non-letters etc) and the requirement to change them several times a year. A first instinct to make life easier is to write down the passwords in a conspicuous place, for example, on a post-it-note clearly attached to the side of the monitor or on a desk calendar. This option is a major security breach, as the identity has now been uncovered and available for unauthorized users to apply in any way they choose (Boatwright & Luo 2007).

Biometric is the field of technology devoted to verification or authentication of individuals using biological traits. Biometric techniques use unique personal features derived from the biometrics of the user to verify the identity claimed. Users have no longer worries about remembering PINs, passwords or pass phrases. Nor they have to worry about forgetting or losing smart cards. Unlike passwords, biometric for the most part is permanent and cannot be easily changed. This type of authentication cannot be lost, forgotten or easily shared with others as can be done with other objects used for traditional authentication because people will always have their physiological or behavioural characteristics with them. As a result of utilizing biometric authentication there is no human resources labour expense associated with password resets due to lockouts or expiration, thereby decreasing a significant percentage of system management costs (Boatwright & Luo 2007). Because of these advantages, biometric authentication methods have become the cutting edge and also the future of superior security.

1.1 Problem Statement

Biometrics could virtually identify individuals with high level of certainty. The prime disadvantage of biometric recognition systems, as previously mentioned, is that if a template is mishandled into wrong hands it could not be easily recalled. As a result, high level of security for stored biometric templates is utterly essential. Conventionally, in most existing biometric recognition systems, at the enrolment stage, the biometric templates such as fingerprint, voice, etc., are captured and banked on a central server. Whenever an input biometric data is captured for detection, it is sent to the server and the handling and matching steps are processed right at the server. In this situation, the safety of the invaluable biometric information is not assured and security attacks could happen during the communication phase or else on the server itself. In an attempt to deal with this issue, embedded biometric recognition systems relocate the signal processing and matching stages from the server into the embedded device and the attained result is communicated with the server. This technique prevents the attacks on communication and the server. In this method there is no need for storage of the biometric templates on several servers for a number of applications. Nonetheless, mishandling the plaintext template stored in the embedded device could happen fairly easily. So, protecting the plaintext biometric template is the main challenge in any biometric authentication system. Therefore, there is a need to use an appropriate cryptographic framework to protect the plaintext biometric template. However, all existing cryptographic algorithms for shielding the information rely on protection of the access key. Therefore, maintaining the secrecy of the access key is the main problem in any cryptographic framework.

1.2 Research Question

Based on the problem described above, the question of this study is the following:

- ➢ How to enhance the process of protecting the users' unique biometric template by cryptographic framework while utilizing the biometric features as a part of some disguising techniques to maximise the secrecy of the cryptographic key?

  By this approach, the biometric features themselves are used to disguise the access key, which in turn, protects the user's biometric template from unauthorised access.

1.3 Research Aims and Objectives

Based on the literature review, the aim of performing scientific research into Biometric technologies in a digital authentication system is to create acceptance for, and increase the quality of, biometric based authentication methods, with the intention of meeting the trust and security requirements in secure systems. Therefore, the objectives of this research can be summarized as follows:

- To perform a thorough review of the biometric techniques and the merging of biometric and cryptography.
- To identify and produce analysis of a biometric technique for an authentication in a secure system.
- To provide a better understanding of the relationship between biometric features extraction techniques and security approaches for cryptographic framework.
- To select and implement at least one promising and advanced biometric and cryptography merged method in order to understand its difficulties and room for improvement. In this research, fingerprint biometrics is ultimately selected and focused on, for its advantages against other biometrics. This will be inspected in more details in next chapter.
- To improve upon a building block in identification and authentication processes. Accordingly, a novel method in fingerprints curve extraction is

proposed which improves upon current methods on computation while preserving the required precision.

1.4 Research Plan

For any research project to be successful, an appropriate plan is required. Therefore, based on the main proposed objectives, an appropriate approach for this project articulates over four phases to include:

Phase 1:

- Literature review; access to literature resources:
    - ➢ UTS library
    - ➢ Digital libraries (ACM, IEEE, Springer link and etc.)
- Problem(s) definition

Phase 2:

- Solution proposition
    - ➢ Critical analysis of other researchers' work to come up with an original or new idea
    - ➢ Address the problem (s) with a reliable solution
    - ➢ Solution evaluation and enhancement

Phase 3:

- Implementation
    - ➢ Evaluate the implemented method on public data sets

Phase 4:

- Feasibility and performance enhancements
    - ➢ Using different techniques and methods

CHAPTER 2 AN INTRODUCTION TO BIOMETRICS AND CRYPTOGRAPHY

2.1 Digital Identity

Digital identity can be defined as the electronic representation of the information known about a specific individual or organization. Such information can be used for a variety of purposes, ranging from allowing one to prove his/her identify and/or to establish permissions (Jones, Anton & Earp 2007).

Qualities for an ideal identity has been described as: (Isalam 2007)

- Uniqueness: It refers to only one individual
- Consistency: Interpreted in the same way by all parties
- Persistency: Identity Information remains constant all the time
- Verifiability: It is easy to identify
- Trust: It refers to the ease with which trait of an individual can be imitated using artefacts.

Based on the quality definition for ideal identity qualities, biometrics can be defined as the foundation of a highly secure identification and personal verification solutions (Bala 2008).

2.2 Biometrics

The international standards committee on biometrics (ISO/IEC JTC1 SC37) define it as "automated recognition of individuals based on their behavioural and biological characteristics." Furthermore, the International Biometrics Society defines biometrics as the "application of statistical and mathematical theory and methods in the biosciences." (Wayman 2008).

Biometric characteristics can be divided into three main classes: (Bhattacharyya et al. 2009)

- Physiological biometrics are physical characteristics, which refer to inherited traits. They are related to the shape of the body and thus they vary from person to person. Some of the physiological features measured include an individual's fingerprints, face and iris.
- Behavioural biometrics is related to the behaviour of a person and some of them are key stroke dynamics and hand writing.
- Combination of physiological and behavioural biometrics (ex. Voice)

2.3 Biometric Strength

The advantage claimed by biometric authentication is that they can establish an unbreakable one-to-one correspondence between an individual and a piece of data. In addition, by using biometrics it is possible to confirm or establish an individual's identity.

Biometric authentication is highly reliable, because physical human characteristics are much more difficult to forge. Utilized alone or integrated with other technologies such as smart cards, encryption keys and digital signature, biometrics is anticipated to pervade nearly all aspects of the economy and our daily lives (Bhattacharyya et al. 2009).

2.4 Types of Biometric Methods

In case of information security, biometrics is used as a method of authentication to verify that a user is the person in the claimed identity, before access to the system is granted. The leading biometric technologies are: (Isalam 2007; Sabena, Dehghantanha & Seddon 2010; Wayman 2008)

- Facial recognition (physical)
- Fingerprint (physical)
- Iris or retina recognition (physical)
- Hand geometry (physical)
- Palm print Pattern (physical)
- Voice recognition (physical and behavioural)
- Signature verification (behavioural)

Emerging Biometric technologies are:

- Gait recognition (behavioural)
- Thermal facial recognition (physical)
- Earlobe recognition (physical)

Table 1, based on literature review, illustrates the more common biometric technologies and their performance in terms of accuracy, ease of use, user acceptance and cost.

Table 1: Characteristics of the most popular biometric technologies

Biometric	Accuracy	Ease of use	User acceptance	Cost
Iris	High	Medium	Low	High
Face	Medium	High	High	Low
Fingerprint	High	High	Medium	Low
Hand geometry	Medium	High	Medium	High
Voice	Medium	High	High	Low
Signature	Medium	Medium	Medium	High

2.4.1 Iris

The iris pattern for each eye is individually unique. Even the irises of identical twins are different. Compared to other biometrics, the iris is protected from the external environment behind the cornea and the eyelid. It is not subject to the effects of aging and the small-scale radial features of the iris remain stable and fixed from about one year of age throughout life (Bhattacharyya et al. 2009).

Iris recognition methods use the iris of the eye which is the coloured area that surrounds the pupil. Iris patterns are unique and are obtained through video based image acquisition systems. Each iris structure features a complex pattern.

Localization of the iris is an important step in iris recognition because, if done improperly, resultant noise (e.g. Eyelash) in the image may lead to poor performance (Bhattacharyya et al. 2009).

Iris technology cannot be easily artificially duplicated because of its unique properties. There is closed connection from iris to the human brain and it is said to be one of the first parts of the body to decay after death. Therefore, it is very difficult to create an artificial iris to fraudulently bypass the biometric systems if the detection of the live iris is working probably. Iris recognition has the potential to be an obstacle or excluder if improperly configured or installed without consultation and guidance from disabled individuals, e.g. blind people, people with Nystagmus (tremor of the eyes), etc. (Bhattacharyya et al. 2009).

Data collection is a very critical issue on any iris recognition system, since it is not easy to capture a valid image of such a small region. If the user just only be asked to stand in front of the camera, by using an artificial vision technique, it is possible to locate the position of the eye and then capture an image. However, this approach is very complex to implement. The complexity of the data collection can be reduced if the user puts his/ her eye in front of the camera. Nonetheless, it has low user acceptance rate since it requires considerable user involvement. For example, it cannot work accurately without support and assistance for disabled or blind individuals. Also, positioning the camera to catch the sample may sometimes be a challenge because it must be positioned accurately (Boatwright & Luo 2007; de Luis-García et al. 2003). These difficulties generally prevent iris recognition from being broadly acceptable in commercial applications, despite its high accuracy. Furthermore, it needs special costly hardware which is not economical (Bhattacharyya et al. 2009; Isalam 2007; Wayman 2008).

2.4.2 Face

Although face recognition cannot be as precise as Iris identification method, it is the most promising technique due to its non-intrusiveness and wealth of features. The image of a human face can be obtained through a simple and natural approach. Face authentication possesses the following characteristics that other biometrics lack:

1. Ability to capture facial images from a distance,
2. Special actions are not always required for authentication
3. It is expected to deter crimes since it is possible to record the captured images,

In addition, a human does not need any device to verify who the person is (Sabena, Dehghantanha & Seddon 2010).

The equipment for collecting human face images is increasingly becoming economical and widespread. Because of its ease in sampling and its recognition without contacting the target, it is easily accepted by end-users. However, the recognition algorithms impose a number of limitations on how facial images are obtained. They also have difficulty in identification under different lighting conditions (Chenggang & Yingmei 2009).

2.4.3 Fingerprints

Among all the biometric techniques, fingerprint-based recognition is the oldest method which had been used for identification as old as 3100 years ago when handprints were used by the prehistoric men as signature for their paintings (Bala 2008; Sabena, Dehghantanha & Seddon 2010). Furthermore, it can be one of the best-known biometrics, because of its common and widespread application in forensic and law enforcement cases (Tistarelli, Bigun & Grosso 2005).

Fingerprint–based identification is popular for individual identification. This is because fingerprints are considered to be unique in view of the fact that the pattern of ridges and valleys of each fingerprint is unique and it will not change by age (Mil'shtein et al. 2008). Due to the size reduction of the sensors in fingerprint verification systems, it is becoming more popular, cheaper and acceptable (Wayman 2008).

However, people are very concerned with their personal privacy. For example, some individuals are hesitant to provide their fingerprints as they relate it to criminality and find the procedure of fingerprinting quite invasive (Boatwright & Luo 2007).

2.4.4 Hand Geometry

Hand geometry is quite simple, fairly easy to use and environment effects, for instance dry weather, or individual irregularities, for example dry skin do not generally have negative effects on the identification accuracy of hand geometry-based authentication systems. Nevertheless this system requires special hardware to use for authentication which is fairly large and cannot be embedded in many devices. In addition, individual hand features are generally not descriptive enough for verification (Isalam 2007; Wayman 2008).

2.4.5 Palm Print

Palm print pattern is one of the newer types of biometric authentication in which the veins in a person's hand are scanned with infrared lighting. This particular method is quite unique and has proven to be very reliable as no two people have the same vein patterns. Other than size, palm print patterns do not change over the course of one's lifetime. As a result, palm print re-imaging does not become an issue here. However, palm print authentication has not gained wide acceptance by the public, as it is still too new in the biometrics field (Boatwright & Luo 2007).

2.4.6 Voice

Voice recognition is also a natural way for solving identification and verification problem. With largely present telephone networks and inexpensive microphones on computers that are connected to the World Wide Web, user recognition through speech has become an easy solution and reasonably acceptable by end-users, and also an economical solution for business use (Shukla & Tiwari 2008).

The components of voice are broken down into three categories called phonemes pitch, intonation, and pronunciation of an individual's voice. These three elements determine the uniqueness of the sound of each person's voice. The voice is sensitive to its background and environment noise, therefore it is considered to be a physiological and behavioural biometric. Unlike many other biometrics, the voice can change to varying degrees caused by illness, emotions, aging and one's environment. Spoofing attacks on voice biometrics are generally accomplished by playback of a phrase that the spoofer is simply attempting to impersonate an authenticated user's voice. In addition, qualities of the microphone and the communication channel could affect the voice recognition outcome (Boatwright & Luo 2007; Wayman 2008).

2.4.7 Signatures

Signature verification recognizes the distinct behavioural characteristics of a user signature consisting of pen pressure, speed, stroke, shape, and timing information (five dimensions). Because it is a behavioural biometric, signature may change over a period of time and could also be influenced by physical and emotional circumstances of the signer. Furthermore, skilled forgers may be able to duplicate signatures; therefore, a very expensive, five-dimensional pen is required for higher accuracy (Isalam 2007; Wayman 2008).

2.5 Biometrics System Functionality

There are two ways that biometrics can be differentiated according to their functionality: identification and authentication (Bhattacharyya et al. 2009; Boatwright & Luo 2007).

- **Biometric Identification:** The process of matching an individual to one of the samples from large sets of system users is called biometric identification. In other words, an identification system identifies an individual by searching all the users in the databases for a match. It tries to identify the person by asking "who is this person?"
- **Biometric Authentication (or Verification):** It is a method utilized by matching an individual's genetic traits or behavioural characteristics with data that have previously been captured, enrolled into a template and stored in a system database or on a token. In other words, the process of verifying that the individual is who he or she claims to be is biometric authentication. In addition, it tries to affirm or deny a person's claimed identity by asking "is this person whom he claims to be?" Authentication is one-to-one comparison of the biometric sample with the record held for the particular user. So the techniques used for biometric authentication

has to be stringent enough that they can employ both these functionalities simultaneously.

2.6 Biometric System Structure

Every biometric system consists of four basic modules as follows (Sabena, Dehghantanha & Seddon 2010):

- **Enrolment unit:** the enrolment module registers individuals into the biometric system database. During this phase, a biometric reader scans the individual's biometric characteristic to produce its digital representation.
- **Feature extraction unit:** this module processes the input sample to generate a template, which is then stored in a central database or a smart card is issued for the individual.
- **Matching unit:** this module compares the current input with the template. If the system performs identity verification, it compares the new characteristics to the user's master template and produces a score or match value (one-to-one matching). A system performing identification matches the new characteristics against the master templates of many users resulting in multiple match values (one-to-many matching).
- **Decision maker unit:** this module accepts or rejects the user based on a security threshold and matching score.

2.7 Biometric Systems Accuracy and Performance

The performance and accuracy evaluation of biometrics systems consider two types of errors known as acquisition and matching errors, described in the following (Sabena, Dehghantanha & Seddon 2010):

- The acquisition errors would be evaluated based on the following two benchmarks:
    - Failure to Capture Rate (FTC): Proportion of attempts for which a biometric system is unable to capture a sample of sufficient quality.

- Failure to Enrol Rate (FTE): Proportion of the user population for which the biometric system is unable to generate reference templates of sufficient quality.

- The matching errors would be evaluated based on following two benchmarks:
  - False Rejection Rate (FRR): It is defined as falsely rejecting a genuine user (client).
  - False Acceptance Rate (FAR): It is defined as falsely accepting an impostor

Note that FAR and FRR are also often denoting as False Match Rate (FMR) and False Non-Match Rate (FNMR). These benchmarks are desirable since they are not application dependent. There is a firm trade-off between FMR and FNMR in every biometric system. In the other words, both FMR and NFMR are functions of a system accuracy threshold *t*. If *t* is decreased to make the system more tolerant with respect to input variations and noise, then FMR increase; conversely, if *t* is raised to make the system more secure, then FNMR increase accordingly. Above and beyond FMR and NFMR, a “compact” value is commonly used to summarize the accuracy of a verification system: the Equal-Error Rate (EER) denotes the error rate at the threshold *t* for which false match rate and false non-match rate are identical; FMR=NFMR (Tistarelli, Bigun & Grosso 2005).

In both authentication and identification modes, biometric decision is never 100% perfect. Because of changes, such as environment, lighting or user positioning, every time a biometric is captured, the characteristics obtained are most likely to be a little different. Even then, the ideal biometric authentication system will try to maintain the rates as low as possible and present the characteristics of user friendliness, reliability, fast operation, accuracy and low cost (Sabena, Dehghantanha & Seddon 2010).

2.8 Biometric Issues

As it is with any technology, biometric methods also have their shortcomings and they are not flawless. We can generally divide issues in Biometric into five types, namely: reliability, privacy, user acceptance, technical issues, and cost (De Luis-García et al. 2003; Sabena, Dehghantanha & Seddon 2010). Following, each issue is discussed in further detail.

2.8.1 Reliability

Correct authentication cannot be guaranteed by using biometric techniques. A false identification or rejection can happen because of sensor or environmental noise, limitations of processing techniques and different algorithms, and the most important factor, variability in both biometric features and their presentation. Furthermore, the precision of a specified biometric implementation is sensitive to the object population. To apply a successful identification application, it is critical to discover and evaluate the biometric technologies in the context of the target application and the object population. It can be said, reliability issues are especially significant for large-scale systems since otherwise exceptional accuracy may become noticeably insufficient.

2.8.2 Privacy

Biometric information typically is a sensitive data. If the information is stolen or leaked by any means and falls into wrong hands, it has the potential of having an immense impact on the individual. The data is characteristically unique and of high quality. It is about an individual's physical unique features which need to be handled cautiously to make certain it is protected. This sensitive information needs to be encrypted, securely stored, only accessible to authorized individuals and destroyed when no longer needed for the purpose under which it was gathered. There are two types of privacy concerns regarding biometrics; personal privacy and information privacy. Personal privacy is about the uneasiness of the individuals when they come across this technology. Some individuals might find biometrics information gathering invasive, unpleasant or troubling. Information privacy is considered as of highest importance, as it is about protecting the sensitive biometric data such as individual’s genetic background, health record and age against unauthorized collection, storage and exploitation.

2.8.3 User Acceptance

Note that a biometric identification system, which stores the biometric templates in a central database or provides the users’ biometric insecurely, may not be acceptable to a user. Evidently, a user's biometric data could be used for ill purposes if the biometric is obtained by a malicious unauthorized individual. A user's biometric can supply information which a user may not desire to provide voluntarily. For example, a fingerprint reading could be used for law enforcement purposes, an eye scan (retinal or iris) may be used to detect medical conditions and genetic information could potentially be used to detect medical preconditions. Factors that make systems undesirable include concerns:

1- about the safety of (long-term) use of the equipment (e.g. many users have concerns about iris “scanners” damaging their eyes, and some are concerned that criminals may cut off their fingers or remove their eyes to access their accounts).
2- About the trustworthiness of identification (e.g. being mistaken for a terrorist).
3- That data might be used for other than the intended purpose (e.g. health diagnostics, tracking, and direct marketing).
4- About the capability or commitment of those responsible for holding the data, to keep them secure from internal and external attackers (malicious employees, hackers, etc.).

2.8.4 Technical Issues

There are eight points at which a biometric system can be attacked. The greatest publicity is received for fake biometric attacks by impostors. However, all other types of attacks require some form of access to the biometric processing systems and conceivably correspond to a more serious risk. Below is a description of the eight potential sources of attack on biometric authentication systems:

1- Attacking the sensor: In this type of attack a forged biometric such as a fake finger or image of the face is presented at the sensor.
2- Resubmitting previously stored digitized biometric signals: In this mode of attack a data related to the presentation of biometric is captured and replayed. Once system is accessed, false data stream is injected between the sensor and the processing system to ease future access privileges.
3- Overriding the feature extractor: This attack interferes with the feature extraction routines to supply false data for further processing. This attack can be used to disable a system and create a denial of service (DOS) attack.
4- Tampering with the biometric feature representation: In this type, after the features have been captured from the input signal, they are swapped with a synthesized feature set of choices given that the template is known by the attackers. Often the two stages of feature extraction and matcher are inseparable and this mode of attack is extremely difficult.
5- Corrupting the matcher: The matcher component is overridden or disregarded and swapped with a match. Also, altering the system tolerances in feature matching, in particular the false acceptance rate (FAR), can cause the system to accept poor quality or false data.
6- Tampering with the stored templates: The database of enrolled templates is available locally or remotely. This database can also be distributed over several servers. The stored template attacker tries to alter one or more templates in the database which could cause at least denial of service for the corrupted templates.
7- Attacking the channel between the stored template and the matcher: The templates from the stored database are sent to the matcher through a channel which could be tapped in to change the contents of the templates before they safely arrive in the matcher.
8- Overriding the final decision: This attack disregards any processing and overrides the decision data or forces a false acceptance between the

system and the end device. The end device could, for instance, be a door lock or a cash dispenser.

2.8.5 Cost

When implementing a new technology, cost is always a key issue. People frequently focus uniquely on the cost of sensor hardware and associated software, when accounting for the cost of a biometric system. However the actual cost of employing any biometric technology goes far beyond these basic elements. In addition to the above basic costs there will be other expenses associated with installation, integration, administration, user education, data collection, and also system maintenance (Bala 2008).

2.9 Cryptography

Cryptography is the practice and study of hiding information. Cryptography as it has been shown on Figure 1, refers almost entirely to encryption; the process of converting ordinary information, i.e. plain text, into meaningless data, i.e. cipher text (Stallings 2003). Decryption is the reverse, moving from meaningless cipher text to plaintext.

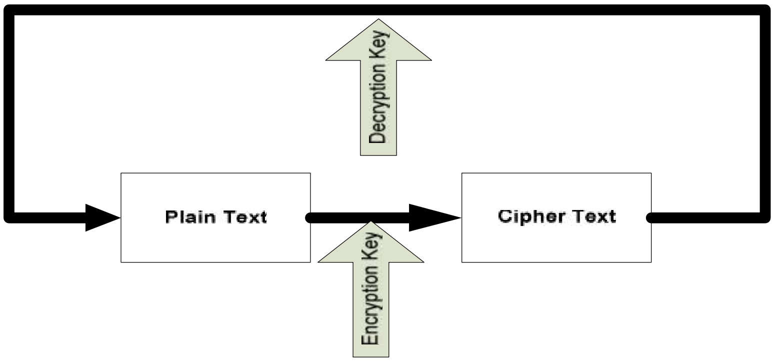


Figure 1: Block Diagram of a Generic Cryptography Framework

In traditional cryptography the encryption key(s) maps the plain text to essentially a sequence of random bits, which can only be mapped back to the plain text using the appropriate decrypting key(s). Without the knowledge of the correct decrypting keys, the exchange of cipher text to the plain text is infeasible considering time and cost limitation (Stallings 2003). Hence, the cipher text is secured and even if an attacker obtains the cipher text, he cannot extract useful information from it. Here, the plain text can be any data that needs to be stored or transmitted securely e.g. biometric template information, email communication, secret cryptographic keys, etc. Cryptography is used in applications such as the security of ATM cards, computer passwords, and electronic commerce, which all depend on cryptography.

In following sections, cryptanalysis (which basically is cryptography in reverse) cryptographic schemes and cryptographic schemes shortcomings are explained.

2.9.1 Cryptanalysis

The process of trying to discover the ordinary information or secret key or both is known as cryptanalysis. The strategy used by the cryptanalyst depends on the nature of encryption scheme and the information available to the cryptanalyst. Brute force and dictionary attack are some of the most well-known attacks in this area (Stallings 2003).

In general it has been assumed that cryptanalyst does know the algorithm used for encryption. Therefore, an encryption format is said to be computationally secure if the following criteria are met:

- The cost of breaking the cipher exceeds the value of the encrypted information.
- The time necessary to break the cipher goes beyond the valuable lifetime of the information.

2.9.2 Cryptographic Schemes

Cryptography not only protects data from theft or alteration, but can also be used for user authentication. There are, in general, three types of cryptographic schemes typically used to achieve these goals, namely;

1. Secret key (or symmetric) cryptography,
2. Public-key (or asymmetric) cryptography
3. Hash functions.

In all cases, the initial unencrypted data is referred to as plaintext. It is encrypted into cipher text, which will in turn (usually) be decrypted into usable plaintext (Menezes, Oorschot & Vanstone 1996). A single key is used for both encryption and decryption in secret key cryptography and two keys are used in public key cryptography.

A hash function uses a fixed-length value computed from the plaintext that makes it impossible for either the contents or length of the plaintext to be recovered. Each cryptography scheme is optimized for some particular application. Hash functions, for example, are well-matched for ensuring data integrity because any change made to the contents of a message will result in the receiver calculating a different hash value than the one placed in the transmission by the sender.

Secret key cryptography, on the other hand, is ideally suited to encrypting messages. The sender can generate a session key on a per message basis to encrypt the message and the receiver, of course, needs the same session key to decrypt the message. Key exchange is a key application of public-key cryptography. Asymmetric schemes can also be used for non-repudiation and if the receiver can obtain the session key encrypted with the sender's private key, then only this sender could have sent the message.

2.9.3 Cryptographic Algorithms Shortcomings

Current cryptographic algorithms namely, Advanced Encryption Standard (AES), Data Encryption Standard (DES) and RSA have a very high proven security but they suffer from the key management problem. All these algorithms fully depend on the assumption that the keys will be kept in absolute secrecy.

Another limitation of these algorithms is that they require the keys to be very long and random for higher security, which makes it impractical for users to memorize the keys. As a result the cryptographic keys are stored securely for instance in a computer or on a smart card and released on some alternative authentication mechanism. If this authentication succeeds, keys can be used in encryption/decryption procedures.

The most popular authentication mechanism used for key release is based on passwords, which are again cryptographic key-like strings but simple enough for users to remember. Hence, the plain text protected by a cryptographic algorithm is only as secure as the password (weakest link) that releases the correct decrypting keys (Cachin et al. 2004). In later chapters our focus would be on our choice, fingerprint biometric modality, and the techniques of employing it.

CHAPTER 3 FINGERPRINT BIOMETRIC MODALITY

3.1 Fingerprint Characteristics

As a result of our review and considering pros and cons of available biometrics, the fingerprint biometric modality is selected to study and develop a secure authentication system. Fingerprint is considered to be the best choice for most applications, from network security systems to compact devices, due to its accuracy, speed, reliability, non-intrusive interfaces, and cost effectiveness (Singla & Saini 2009).

A fingerprint is the pattern of lines, which are called ridges and the spaces between individual ridges which are referred to as valleys on the surface of a finger (see Figure 2) (Maltoni et al. 2003).

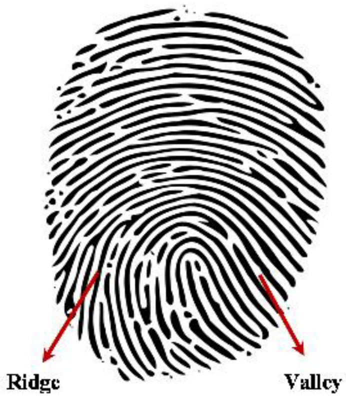


Figure 2: Ridges and Valleys in a Fingerprint Image

Ridge details are generally clustered into two different groups which are the overall global ridge flow pattern and minutiae points. At the global level, ridges often run smoothly in parallel but exhibit one or more regions where they assume distinctive shapes (characterized by high curvature, frequent ridge terminations, etc.). These regions, called singularities or singular regions may be broadly classified into three typologies: loop, delta, and whorl (see Figure 3).

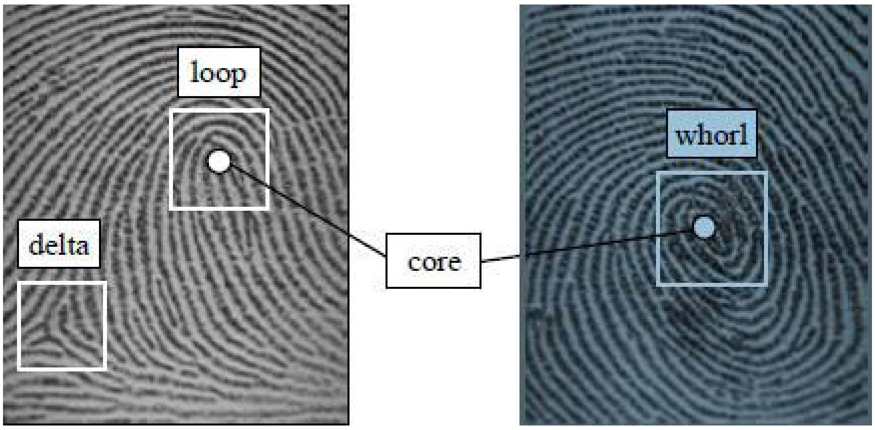
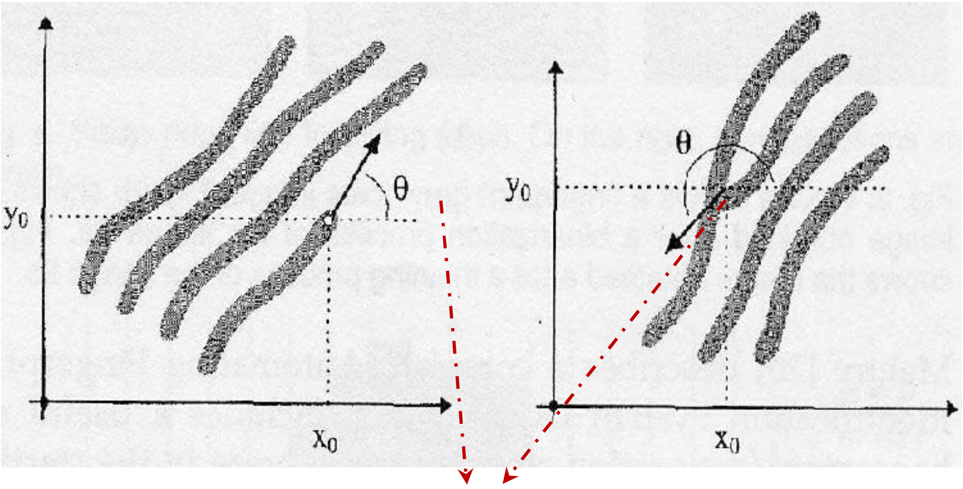


Figure 3: Singular Regions in a Fingerprint Image

At the minutiae points, a ridge can either come to an end, or it can divide into two ridges. As it has been illustrated in Figure 4, the location where a ridge comes to an end is known as a termination, and the location where a ridge divides into two separate ridges is called a bifurcation.

Terminations **Bifurcation**



Minutiae' Orientation or Angle

Figure 4: Basic Types of Minutiae Points

A cut or burn to a finger does not affect the fundamental ridge structure, and the unique pattern will be reproduced when new skin grows. Therefore, fingerprint authentication is one of the most important biometric technologies since, except in rare cases of extreme accidents causing irreversible damages; the fingerprints stay unchanged during human lifespan and so does the uniqueness of each individual’s fingerprints. The uniqueness of the fingerprint can be identified by the characteristics and relationships of bifurcations and endings in ridges or valleys.

In order to compare two fingerprints, a set of invariant and discriminating features are extracted from fingerprint image. Terminations and bifurcations are the two basic types of minutiae, which are the points of interest within a fingerprint. But there are other types of minutiae that are to some extent more complicated combinations of terminations and bifurcations. For example, a lake is simply a sequence of two bifurcations in opposing directions, while an independent ridge features two separate terminations within a close distance (Singla & Saini 2009).

3.2 Fingerprint Classification

The identification of a person needs a comparison of his/her fingerprints with all the fingerprints in a database. In a large scale database, the fingerprint identification process typically has an unacceptably long response time. Therefore, applying an intermediate classification approach can greatly reduce the number of comparisons during the fingerprint retrieval and accordingly decrease the response time of the identification process.

Fingerprint classification refers to the problem of assigning a fingerprint to a class in a consistent and reliable way. There is no limitation in using any of local or global features for either matching or classification. Although fingerprint matching is usually performed according to local features (e.g., minutiae), and fingerprint classification is generally based on global features, such as global ridge structure and singularities (Maltoni et al. 2003). The most widely used fingerprint classification method is based on Henry’s classification which consists of eight classes: Plain Arch, Tented Arch, Left Loop, Right Loop, Plain Whorl, Central-Pocket Whorl, Double Loop Whorl and Accidental Whorl. Figure 5 shows basic categories of fingerprint classes (Ahmad & Mohamad 2009).

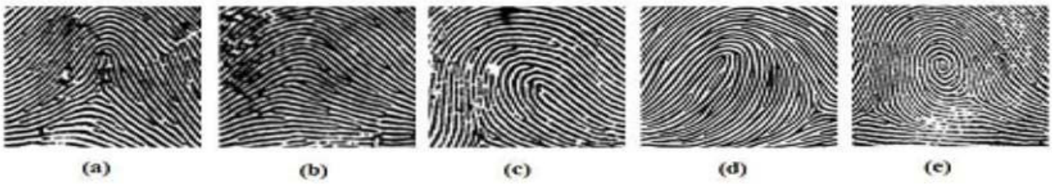


Figure 5: Basic categories of fingerprint classes: (a) Tented Arch (b) Arch (c) Right Loop (d) Left Loop (e) Whorl.

Automated fingerprint classification is a challenging and demanding pattern recognition problem due to several reasons. One of the factors is in relation to large intra-class variation and small inter-class variation in fingerprint images. In some cases, prints from one class can have similar appearance to prints from another class and prints from same class can have very different characteristics. Another major challenge is related to the presence of noise in fingerprint images, which makes the classification task even more difficult. Random noise and other effects caused by skin conditions such as dry, moist, dirty, and diseased can cause errors in the fingerprint images. It is an important task in fingerprint classification systems to recover the original ridge patterns, and therefore preprocessing steps are required to enhance the fingerprint image. One more factor concerning fingerprint classification performance is ambiguous fingerprints. There are some fingerprints that are ambiguous and cannot be classified even by a human expert. A fingerprint classification system must devise a method for dealing with these problems, such as having an "anomalies" class or rejecting them outright.

The criteria to select an accurate fingerprint classification technique heavily depend on the number of classes and the natural distribution of fingerprints. Unfortunately, the number of classes used is often small and the distribution of the classes in nature is not uniform and further affected by these ambiguous fingerprints.

The major problem in designing a fingerprint classification system is to determine what features should be used and how these features can classify the fingerprint into their categories. It has been shown that fingerprint classification not only reduces the required number of comparisons of fingerprints, but also improves the overall efficiency of the fingerprint identification system (Ahmad & Mohamad 2009).

3.2.1 Fingerprint Classification Methods

Most of the existing fingerprint classification methods can be coarsely assigned to one of these categories (Maltoni et al. 2003):

- **Rule based:** this is the approach commonly used by human experts for manual classification. In this method, a list of rules or rule base, which is a specific type of knowledge base, devised by human experts, is employed. These methods are attractive in automatic classification because of their simplicity. Yet, some problems arise in the presence of noisy images.
- **Syntactic based:** A syntactic method describes patterns by means of terminal symbols and production rules; a grammar is defined for each class and a parsing process is responsible for classifying each new pattern. In general, due to the great diversity of fingerprint patterns, syntactic approaches require very complex grammars whose inference requires complicated and unstable approaches; for this reason, the use of syntactic methods for fingerprint classification has been almost abandoned, with a few exceptions.
- **Structure based:** Structural approaches are based on the relational organization of low-level features into higher-level structures. This relational organization is represented by means of symbolic data structures, such as trees and graphs, which allow a hierarchical organization of the information. The orientation image is well suited for structural representation: in fact, it can be partitioned into connected regions that are characterized by “homogeneous” orientations; these regions and the relations among them contain information useful for classification. But it is not easy to robustly partition the orientation image into homogeneous regions, especially in poor quality fingerprints.
- **Statistical based:** In statistical approaches, a fixed-size numerical feature vector is derived from each fingerprint and a general-purpose statistical classifier is used for the classification. Some of the most widely

adopted statistical classifiers are: Bayes decision rule, k nearest neighbors, and Support Vector Machine (SVM).

- **Neural network based:** Most of the proposed neural network approaches are based on multilayer perceptrons and use the elements of the orientation image as input features.
- **Multi-classifier approaches:** Different classifiers potentially offer complementary information about the patterns to be classified, which may be exploited to improve performance. Indeed, in a number of pattern classification studies, it has been observed that different classifiers often misclassify differently.

Comprehensive analysis of fingerprint classification methods published over the last 30 years has shown the interest in syntactic approaches in the 1970s/1980s, and the success of neural networks in the 1990s and of multiple classifier systems in recent years.

3.3 Fingerprint Identification and Authentication System

As discussed before, there are two different modes in any biometric system functionalities, namely identification and authentication modes. The fingerprint feature extraction and matching algorithms are usually quite similar for both fingerprint identification and authentication modes. This is because the fingerprint identification mode (i.e., searching for an input fingerprint in a database of N fingerprints) can be implemented as a sequential execution of N one-to-one comparisons (authentication) between pairs of fingerprints (Maltoni et al. 2003).

In general the process of using any type of Fingerprint Authentication System requires several steps as follows (Singla & Saini 2009):

- Image pre – processing
- Feature extraction
- Saving and protecting the template
- Matching

3.4 Fingerprint Images

The main parameters characterizing a digital fingerprint image are as follows (Tistarelli, Bigun & Grosso 2005):

1- Resolution: It specifies the number of dots or pixels per inch (dpi). It allows the extraction algorithms to locate the minutiae in fingerprint patterns.

2- Area: The fundamental parameter of a fingerprint scanner is the size of the rectangular capture area. A larger area means more captured ridges and valleys that result in more distinctive fingerprints. Small-area scanners do not allow a whole fingerprint to be captured to reduce their cost but this causes ensuing difficulties in re-presenting the same portion of the finger, eventually leading to FRR errors.

3- Number of pixels: It can be easily calculated by the resolution (r) and the fingerprint area of height (h) × width (w) inch^2 that results in $rh \times rw$

4- Dynamic range (or depth): It indicates the number of bits used to encode the intensity value of each pixel.

5- Geometric accuracy: This is specified by the maximum geometric distortion introduced by the acquisition device (scanner).

3.4.1 Image Pre-Processing

Automatic and reliable extraction of the minutiae points from the input fingerprint images is a crucial step in automatic fingerprint matching. The reliability of minutiae extraction is greatly affected by the quality of acquired fingerprint images. The quality of captured fingerprint images could have undesirable characteristics such as lack of clarity and contamination during the capture phase caused by the followings (Wu et al. 2007):

1- **Transforming a three-dimensional feature into a two-dimensional image:** The first distortion occurs simply because of the nature of the fingerprint image sensor. The fingerprint is distributed in a three dimensional curved surface of the finger tips but a fingerprint image is a two-dimensional image captured by the flat fingerprint sensor. Therefore a conversion from a three-dimensional surface to a two-dimensional surface occurs. The nature of this distortion depends on the amount of pressure and the angle of the finger and the sensor. This distortion happens differently every time and cannot be controlled or exactly duplicated.

2- **Non uniform touch:** The entire formation of a fingerprint can be captured providing that all the ridges contact the sensor. However that is not usually the case because of dryness, wetness, dirt, etc. Some ridges could be incomplete because of the soft touch and others could be unfinished and not well defined because of the firm touch which will make the fingerprint identification more complex.

3- **Unrecoverable fingerprint:** Major damages to the fingers caused by accidents could change the texture of skin or miss so much of the fingertip to make the fingerprint useless and/or unrecoverable.

4- **Limitations of fingerprint capture sensors:** Each type of fingerprint capture sensors such as light sensors, resistance sensors, capacitance sensors, ultrasonic sensor, thermal sensors, etc. has its own drawbacks which makes capturing a perfect fingerprint complicated.

5- **Noise from the sensor's surface:** The surface of the fingerprint sensor could be stained, scratched, it can shake during the capture, there may be slight imprints left from previous fingerprints, etc. These issues could introduce noise into the captured fingerprints.

As a result it is crucial to pre-process the input fingerprint images to minimize the effect of noise and in order to make certain that an automatic fingerprint identification/authentication system is robust against the quality of the fingerprint images.

An image pre-processing step generally includes different steps namely (Jain, Lin & Bolle 1997; Ratha, Chen & Jain 1995):

- Image Enhancement
- Orientation Field Estimation
- Image segmentation
- Ridge/Edge detection and enhancement

- Image Binarization

Next, each step is fully described.

3.4.2 Image Enhancement

The performance of minutiae extraction algorithms and other fingerprint recognition techniques relies heavily on the quality of the captured fingerprint images. In a perfect captured image, ridges and valleys interchange and flow in locally constant direction. Therefore, the ridges can be easily recognized and minutia points can be accurately established in the image. However, in reality, due to skin conditions, sensor noise and incorrect finger pressure, a considerable percentage of fingerprint images are of poor quality.

The goal of enhancement algorithms is to improve the precision of the ridges structures in the recoverable areas and mark the unrecoverable areas as too noisy for further processing (Tistarelli, Bigun & Grosso 2005). Image enhancement consist of improvement of the visibility and perceptibility of different regions into which an image can be divided and also detectability of the image features within these regions.

3.4.3 Orientation Field Estimation

Fingerprint images can be characterized as an oriented texture pattern. The direction of flow of the ridge structures at each location in the image can be characterized as a two-dimensional vector with unit norm. The orientation field (also called directional field) is defined as the set of orientation vectors for all sites in the image (Rao 1990). The orientation field at [*x, y*] is the angle θ_{xy} that the fingerprint ridges, crossing through an arbitrary small neighbourhood centred at [*x, y*], form with the horizontal axis (Figure 6) (Tistarelli, Bigun & Grosso 2005).

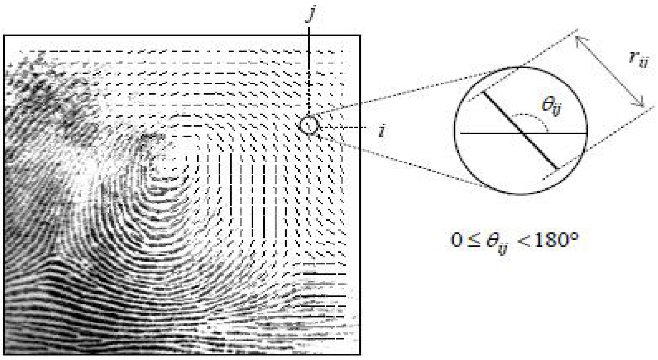


Figure 6 : Orientation Field (Tistarelli, Bigun & Grosso 2005)

A number of different approaches have been proposed for a reliable and fast estimation of the orientation field. They consist of methods based on gradient-based approaches (Lin, Yifei & Jain 1998; Rao 1990; Ratha, Chen & Jain 1995), filter-based approaches (O'Gorman & Nickerson 1989) and Markov random field models (Dass 2004; Zhang, Brady & Smith 2001).

Rao's algorithm (1990) is the most commonly used to compute the orientation field in minutiae (local features) extraction for fingerprint verification (Dass 2004). However, the Markov random fields for orientation field estimation proposed by Dass (2004) is specially designed for singularities (global features) detection in fingerprint classification (Dass & Jain 2004a).

3.4.4 Image Segmentation

Segmentation is the process of sorting out the foreground areas in the image from the background areas. The foreground areas keep up a correspondence to the comprehensible fingerprint area containing the ridges and valleys, which is known as the region/area of interest. The background corresponds to the areas outside the boundaries of the fingerprint area, which do not contain any valid fingerprint information. The task of the fingerprint segmentation algorithm is to decide which part of the image belongs to the foreground and which part to the background. Accurate segmentation is especially vital for the reliable extraction of fingerprint features. Most of the feature extraction algorithms extract a lot of false features when applied to the noisy background area. Therefore, it can be said , the main goal of the segmentation algorithm is to get rid of the background, and thus reduce the number of false features and speed up the computational processes (Maltoni et al. 2003).

Several approaches to fingerprint image segmentation are known from literature. In (Yang et al. 2010), these approaches are divided into block-wise and pixel-wise methods. The first methods divide a fingerprint image into separate blocks of the same size, and then categorize the blocks into foreground and background based on the extracted block-wise fingerprint segmentation features. Pixel-wise methods try to classify pixels to distinguish foreground and background based on the estimation of block-wise fingerprint segmentation features. The commonly used features in fingerprint segmentation consist of grey-level features, orientation features, frequency domain features, and so forth (Alonso-Fernandez, Fierrez-Aguilar & Ortega-Garcia 2005; Yang et al. 2010).

Dolezel et al. (2010), classify the existing segmentation algorithms into three main categories namely, National Institute of Standards and Technology (NIST), Gradient based and Gabor filter based algorithms.

NIST provides implemented algorithms that can be used for fingerprint segmentation. For example, the NIST Biometric Image Software (NBIS) package contains the Segmentor routine, which deals with fingerprint segmentation for fingerprint classification. By using special thresholding based on global and local pixel intensity minimums and maximums massive erosion and edge detection, the Segmentor routine computes the most suitable fixed size rectangle in input fingerprint and declares it as a segmentation result. Other example of segmentation method using by NIST algorithm is segmentation based on NIST fingerprint Image Quality (NFIQ). In this method, first the input fingerprint image quality map is computed using NFIQ algorithm. Then the result image is created by special thresholding, where areas with quality equal or better than the specific threshold are considered as fingerprint area whereas other areas are marked as background.

Gradient based algorithms are block-wise and work based on the different parameters such as gradient intensity, gradient coherence, average gradient on each block, grey intensity mean and variance for segmentation (Alonso-Fernandez, Fierrez-Aguilar & Ortega-Garcia 2005). On the other hand, Gabor filter has the feature of both orientation and frequency selection, so it is particularly appropriate for fingerprint images. It can effectively deal and fix small breaks on ridges. Furthermore, it can remove noises (Wang et al. 2010). Shen, Kot & Koo (2001) have shown that when good quality images are considered, both Gradient and Gabor based algorithms have comparable results. However, they have concluded that Gabor filter based methods are faster.

3.4.5 Ridge/Edge Detection and Enhancement

The ridge/edge detection of the fingerprint images is a very important part of fingerprint pre-processing steps. The most leading property corresponding to ridges in a captured fingerprint image is the fact that grey-level values on ridges reach their local maxima along the normal directions of local ridges (Jain, Lin & Bolle 1997). Because the edge and shape of an image usually have arbitrary directions, it is needed to find some operators which have the same edge and shape detecting ability for arbitrary directions. The basic idea of an edge detection algorithm is to define the edge intensity of the pixels, and extracting the edge point sets by setting the threshold. The most common edge detection methods include binary image edge detection with Robert, Sobel, Prewitt, LOG, Canny and Zero-cross operators. Following, each of these methods are described (Cui et al. 2008).

- **Sobel Operator** detects the edges as the form of filter operator. There are two templates in X and Y directions. These two templates represent gradient operators which are shown in table 2. The Sobel Operator has a good performance on the images with grey gradient and high noise, but the location of edges is not very accurate.

Table 2: Sobel Operators

1	2	1
0	0	0
-1	-2	-1

1	0	-1
2	0	-2
1	0	-1

- **Roberts Operator** is a 2x2 operator. It uses the partial differential operators to detect the edge and calculates the first difference along the direction of 45°. The Roberts operator has a good performance on the images with sharp and low noise, but the edges detected by the Roberts operator are rough, so the location of edges is not very accurate. Its template is shown in table 3.

Table 3: Robert Operator

1	0
0	-1

- **Prewitt Operator** is a weighted average operator. It can suppress the noise, but the pixel average is the same as a Low-pass filter for the images, so locating the edge by using the Prewitt Operator is worse than by using the Roberts Operator. The Prewitt operator has a good performance on the images with grey gradient and high noise, but the edges are wide and have lots of intermittent points. Its templates are shown in table 4.

Table 4: Prewitt Operator

1	0	-1
1	0	-1
1	0	-1

1	1	1
0	0	0
-1	-1	-1

- **LOG Operator** pre-smoothes the images by the Gauss low-pass filter and then finds the steep edges of the image. Finally, binarization is performed by using the zero grey-scale value to generate the closed and connective contours which eliminates all internal points. This operator has two convolution nucleuses and its convolution method is the same as the Sobel operator's convolution method. The LOG operator is sensitive to noise, the edge is wide and has high noise, so it is rarely used to detect the edges, but it is used to determine whether the edge pixels is in the bright areas or dark areas of the image. Its templates are shown in table 5.

Table 5: LOG Operators

0	-1	0
-1	4	-1
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

- **Canny Edge Detection** has a good edge monitoring performance. It first divides the image into several sub-images, and then selects different threshold for each sub-image according to its actual situation, and dynamic threshold dividing is processed for the image according to the threshold selected for each sub-image. The Canny operator is not subject to noise interference and can detect real weak edges. Its advantage is that it uses two different thresholds to detect strong edges and weak edges and a weak edge will be included in the output image only if it is connected to a strong edge.
- **Zero-cross Operator** is a detection method that uses the filter specified to filter the image and then seeks for a zero-cross point as the edge. The Zero-cross operator is not vulnerable to noise interference. The detected edges are thin, the location of the edges is more accurate and therefore the result of the edge detection is of high quality.
- **Binary Image Edge Detection** algorithm is an arbitrary pixel (i,j) and its connection with other eight neighbouring pixels points which is shown in table 6.

Table 6: $(\boldsymbol{i},\boldsymbol{j})$ and the Connection with Neighbouring Pixels

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & (i,j) & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdots \begin{bmatrix} 0 & 0 & 1 \\ 0 & (i,j) & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdots \begin{bmatrix} 0 & 0 & 1 \\ 0 & (i,j) & 1 \\ 1 & 1 & 1 \end{bmatrix} \cdots \begin{bmatrix} 0 & 0 & 1 \\ 1 & (i,j) & 1 \\ 1 & 1 & 1 \end{bmatrix} \cdots \begin{bmatrix} 1 & 1 & 1 \\ 1 & (i,j) & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

According to the definition of the edge, from table 6 it can been seen that if the value of (i,j) is 1, only the last case in figure 5 is a non-edge point.

That is judging all 8 pixels one by one, if there are one or more values of the eight adjacent pixel points that are equal to the opposite of (i, j) then this point is an edge point. Simply, when all eight adjacent pixel values are the same as the value of (i, j), this point is a non-edge point. The binary image edge detection algorithm is simple but it can detect the edges of the image accurately and the processed images do need to be thinned.

Each method has its advantages and disadvantages and the appropriate algorithm should be selected according to the characteristics of the image for optimal performance.

3.4.6 Image Binarization

Binarization is the process that converts a grey-scale image into a binary image. This would improve the contrast between the ridges and valleys in a fingerprint image and consequently would facilitate the extraction of fingerprint features. Most fingerprint feature extraction algorithms operate on binary images where there are only two levels of interest; black pixels that represent ridges and white pixels that represent valleys (Ratha, Chen & Jain 1995).

3.5 Fingerprint Matching Techniques

Fingerprint matching algorithms intend to evaluate two fingerprint images and generate a similarity score corresponding to the probability of the two fingerprints to be the same. The accuracy of an automatic fingerprint authentication system is notably affected by its fingerprint matching algorithm (Sen, Wei & Yang Sheng 2002).

Approaches to fingerprint matching can be roughly classified into the following two major categories:

- **Correlation (graph)-based matching:** Two fingerprint images are placed over one another and the correlation between the corresponding pixels is computed for different alignments (e.g., various displacements and rotations).

- **Minutiae-based matching:** This is the most popular and widely used technique which is similar to the technique used by fingerprint assessors for fingerprint assessment. Minutiae are extracted from the two fingerprints and stored as sets of points in the two dimensional plane. Minutiae-based matching essentially consists of finding the alignment between the template and the input minutiae feature sets that result in the maximum number of minutiae overlap.

The minutiae-based matching is well accepted for the modern embedded fingerprint recognition systems. The reason is that the minutiae of the fingerprints are commonly considered to be the most unique and reliable features of fingerprints. Also, using this method, the template size of the minutiae information of a fingerprint is greatly smaller and the processing speed is much higher than that of graph-based fingerprint matching. In order to save memory and energy in embedded devices these advantageous characteristics are very crucial. To date, a great deal of research has been performed on minutiae-based fingerprint matching. Some methods exploit the local structure of the minutiae to express the attributes of the minutiae set. This technique is fast (computationally low-cost) and is robust against rotation and partial prints. However, the local structure typically is less distinctive because it only corresponds to parts of the entire minutiae set. Fingerprints of different fingers of the same person might have some similar local structures by chance. Then again separate prints from the same finger may only have very few similar structures as a result of the presence of false minutiae and the lack of genuine minutiae. Graph-based matching algorithms identify the outline of the ridges attached to minutiae. This could enhance the system accuracy but it will generate larger template sizes since the connected ridges for each minutia need to be saved as well (Nikam & Agarwal 2008). Some works mix the local and global structures. The local structure is employed to find the correlation between two minutiae sets and improve the reliability of the global matching. The global formation of minutiae consistently determines the exclusivity of a fingerprint (Yang & Verbauwhede 2003). The minutiae based techniques are so developed and mature that in the present market the prime or possibly the exclusive fingerprint features in use are minutiae points (Hu 2008). Accordingly, we have utilized minutiae based algorithm in our implementation.

3.5.1 Minutiae-Based Methods

Minutiae are extracted from the two fingerprints and stored as sets of points in the two-dimensional plane. Most common minutiae matching algorithms consider each minutia as a triplet m = $\{x, y, \theta\}$ that indicates the (x, y) minutia location coordinates and the minutia angle θ:

$$\text{Template} = \{m_1, m_2, \ldots, m_m\}, \qquad m_i = \{x_i, y_i, \ldots, \theta_i\}, \qquad i = 1 \ldots. m \qquad (1)$$

$$\text{Input} = \{m'_1, m'_2, \ldots, m'_n\}, \qquad m'_j = \{x'_j, y'_j, \ldots, \theta'_j\}, \qquad j = 1 \ldots. n \qquad (2)$$

where m and n denote the number of minutiae in template and input image, respectively.

A minutia m'_j in Input (I) and a minutia m_i in Template (T) are considered “matching” if the spatial distance (*sd*) between them is smaller than a given tolerance r_0 and the direction difference (*dd*) between them is smaller than an angular tolerance θ_0:

$$sd(m'_j, m_i) = \sqrt{(x'_j - x_i)^2 + (y'_j - y_i)^2} \qquad \square\ r_0 \qquad (3)$$

and

$$dd(m'_j, m_i) = \min(|\theta'_j - \theta_i|, 360 - |\square'_j - \theta_i|) \ \leq\ \theta_0 \qquad (4)$$

The tolerances defined by r_0 and $\square_0$ are necessary to manage the unavoidable errors made by feature extraction algorithms and to account for the small plastic distortions that cause the minutiae positions to change.

Aligning the two fingerprints is a necessary and crucial step in order to enable overlapping and matching the corresponding minutiae. Clearly, aligning two fingerprints requires displacement (alteration of x and y coordinates) and rotation (alteration of θ) to be recovered. Also, it could likely involve other geometrical transformations like scale and specific distortion-tolerant geometrical transformations (Tistarelli, Bigun & Grosso 2005).

CHAPTER 4 FINGERPRINT TEMPLATE PROTECTION

Cryptographic systems security depends on the secrecy of the secret or private key. Since passcodes or tokens are easy to be forgotten, lost or stolen, using them to release key and granting access, is not secure and convenient enough for ever increased security requirements of many existing applications. Crypto-biometric systems, that bind a cryptographic key with user's biometric information, are novel means to address these concerns of traditional passcode or token based systems. In this mode, biometric templates become priceless assets to be protected and secured.

Securing the template of a user is a critical issue in every biometric system since it cannot be easily retracted if compromised. For instance, if one finger is utilized as a password, after it is compromised, it cannot ever be used again. Evidently it is not feasible that a fingerprint can be modified, which means it is compromised forever. Therefore, it is essential to have a robust cryptosystem to secure the biometric template in a cryptographic framework. At the same time, one of the main difficulties in the cryptosystem is to uphold the confidentiality of the cryptographic keys.

There are different techniques available to secure both the key and the biometric template. Before storing the biometric template, it could be encrypted by a secret key. Within the matching procedure, immediately after receiving the input query, the system will decrypt the template and carry out the assessment. Yet, more eager attackers may be able to unscramble the secure key and then the template. Evidently, physical realization of an algorithm will supply attackers with significant information such as variations in timing, power consumption and electromagnetic radiation. These might be associated to the internal state and consequently to the secret data. This category of attacks is labelled as Side

Channel Attacks (SCA). The strongest within this category is the Differential Power Analysis (DPA) method that analyses power consumption measurements to find secret keys from tamper resistant devices. DPA uses statistical analysis and error correction to determine the information from the power consumption since it is interrelated to the secure data (Yang & Verbauwhede 2003).

A feasible solution to cope with this issue is to store a noninvertible transformed version of the biometric template during the enrolment stage in the embedded device rather than the original template. At detection phase the input biometric data is encrypted using the same noninvertible transform, followed by the matching in the transformed space. Each application may use a different noninvertible transformation or the same transformation but changing its parameters. As a result the template will only be functional for the application that has produced it. If an attacker could uncover such a biometric template the system would then create a new template with another transformation or even with the same transformation but using different parameters (Yang & Verbauwhede 2003).

Currently, the biometric cryptosystems for protecting a fingerprint template are classified into three main categories as follows (Nandakumar, Jain & Pankanti 2007):

1. **Key Release:** In key release mode, the biometric authentication and the key release mechanism are entirely disconnected. The biometric template and the key are stored separately and the key is released provided that the biometric matching is successful.

   It may be easy to realize a biometric cryptosystem in the key release mode but such a system will not be proper for high security applications since it has two major weaknesses. First of all the biometric template is not safe and second, it is possible to override the biometric matcher because authentication and key release are disconnected.

2. **Key Binding:** In the key binding mode, the key and the template are bound as a whole within a cryptographic framework. Decoding the key or the template without any knowledge of the user‘s biometric data will not then

be possible. Authentication and key release will be carried out in a single step within a crypto-biometric matching algorithm.

3. **Key Generation:** In the key generation mode, the key is not kept in a database and is obtained directly from the biometric data. In biometric-based key generation, the key is calculated directly from the biometric information. Biometric cryptosystems operating in this mode are safer but due to sizable variation in biometric data they are more difficult to realize. For instance, aspects such as translation, rotation and non-linear distortion cause variations in the same fingerprint. In addition, the biometric template is equally vulnerable as with the key release scheme.

4.1 Evolution of Mobile Biometric Template Protection

Traditionally, a cryptosystem based on a mobile biometric authentication has a functional scheme as illustrated in Figure 7. A biometric modality is captured by a biometric reader via online scan and its features are retrieved. Then these features pass into the smart card to go through the rest of the process. Within the smart card, the biometric template, that has been stored previously, will be compared with the retrieved features of the captured biometric modality. In the decision making unit if a yes-matching is made, the previously saved cryptographic key will be released to complete the conventional cryptography applications similar to encryption of messages to communicate with application servers (Jain, Bolle & Pankanti 1999).

In this architecture the embedded biometric authentication section can enhance the overall security of the system. For instance, missing tokens cannot be used directly. However, this would generate a serious security problem if impostors gain physical access to the smart card that embeds a biometric template and a secret key separately. The biometric information is a really sensitive datum because of its uniqueness and its persistency during the life span of an individual. When some biometric information is compromised, it can never be revoked or cancelled. Also, an individual has a limited number of applicable biometrics using current technology. Therefore the protection of the mobile biometric template is a severely critical issue (Cimato et al. 2006).

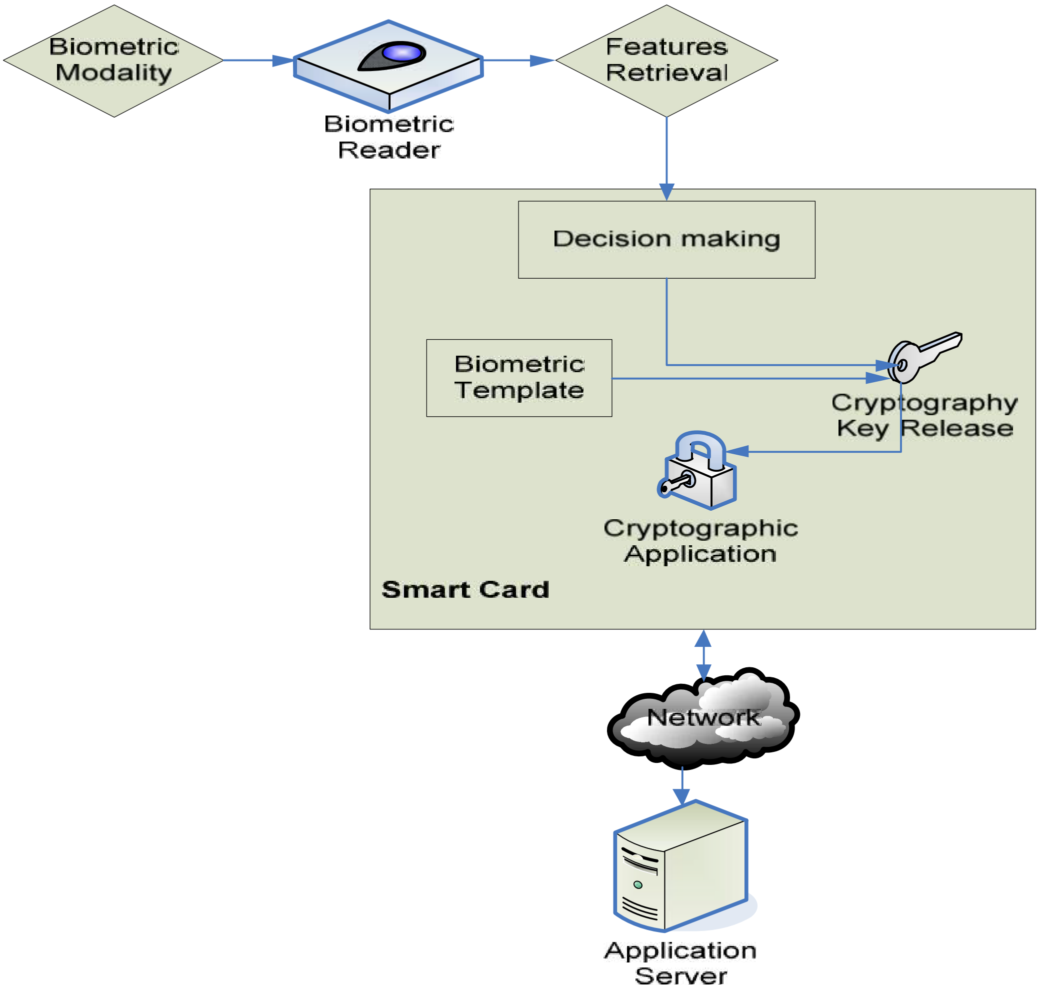


Figure 7: Functional scheme for mobile biometric template authentication based on the key release mode

In general, research efforts in protecting mobile biometric template are classified into three categories, namely; key generation, key binding and irrevocable key generation or cancellable biometric template. Biometric key generation mode normally refers to those schemes that derive cryptographic keys directly from biometrics and in biometric key binding mode, a secret key and a biometric template are bound as a whole within a cryptographic framework (Nandakumar, Jain & Pankanti 2007). The idea behind the cancellable biometric is to use noninvertible transforms to transform biometric features into a new domain where authentication is carried out (Hu 2008). These three modes of biometric cryptosystems are studied specifically for fingerprint modality.

4.1.1 Fingerprint Key Generation

In key generation mode, fingerprint features which are called minutiae are directly mapped into a unique and repeatable binary string and then transformed into a cryptographic key. In the current commercial market, the prime or perhaps exclusive fingerprint feature is the minutia because of the maturity of its related technology and small size of the resulting fingerprint template (Hu 2008). The most significant and attractive characteristic in key generation mode is that no fingerprint template is needed to be stored. But a related problem is the key variety; an individual may wish to have separate keys for his/her bank accounts or for accessing different systems and applications. In this scheme he/she cannot revoke one without affecting the others (Sashank Singhvi et al. 2009). Furthermore, the most difficult problem of this model is that two fingerprint readings are rarely identical, even though they are similar, because of variation in impression conditions, ridge configuration, skin conditions and non- cooperative attitude of a subject (Yuan 2009). Thus it cannot be guaranteed to create the same unique key every time from different fingerprint samples.

Davida, Frank & Matt (1998) made the first contribution in building a framework to create cryptographic keys from biometrics without stored references. As acknowledged by them, this bio-cryptosystem relies on high accuracy of the retrieved biometric features. Although this scheme can be successfully applied to the iris case, it is not directly appropriate for fingerprints since there are many more variations in fingerprint features than in iris features. Farooq et al. (2007) have suggested covering this issue by encoding minutia triangles. This method is required to use a unique personal key for each individual to activate the system. But then the issue is the scheme that this extra key should be managed. Han, Yu & Hu (2005) have proposed an image based fingerprint encryption scheme. A secret key is derived from the captured fingerprint image pixel allocation and some global structure of the fingerprint such as singular points and frequency of ridges in fingerprint. It is shown that the global structure of the fingerprint features consistently determines the exclusivity of a fingerprint (Yang & Verbauwhede 2003). But in this scheme performance testing has not been provided. In addition, this scheme also needs an accurate registration. Later on, Han, Yu & Hu (2005) have recommended a concept of “fictitious triangle”. The idea behind the concept of “fictitious triangle” is that the length parameter of a long line tends to be less affected by distortion and rotation. Likewise in this scheme the false acceptance rate has not been provided. In general, not many practical outcomes on direct key generation from fingerprint template have been reported. This might be because of the huge variations of fingerprint features such as minutiae.

4.1.2 Fingerprint Key Binding

The key binding mode could be a feasible solution since a given cryptographic key is bound with the fingerprint template as a whole within a cryptographic framework (Sashank Singhvi et al. 2009). Fuzzy vault is an example of binding crypto-biometric systems.

Juels & Wattenberg (1999) have proposed the first binding mode, called “fuzzy commitment scheme”, based on the use of error-correcting codes, which considers binary strings where the similarity is measured by Hamming distance. The major issue with this scheme is that it requires the fingerprint representations and features at the enrolment and authentication phase to be ordered but the ordering requirement is unfeasible for fingerprint minutiae especially since it is very common to have a fingerprint with some false and/or missing minutiae.

Juels & Sudan (2002) have introduced fuzzy vault scheme to address the ordering problem among others. In this scheme fingerprint minutiae coordinates *mi* are encoded as the elements in a finite field *F* and a secret key is encoded in a polynomial *f(x)* over *F[x]*. The polynomial is evaluated at the minutiae locations, and the pairs $(m_i, f(m_i))$ are stored along with random (c_i, di) chaff points such that $d_i \neq f(c_i)$. The number of minutiae locations is generally highly redundant for the identification of the polynomial. Given a matching fingerprint, a legitimate user can separate out enough true points from the chaff points to reconstruct *f(x)*, and hence the original secret key. This scheme is called fuzzy since the secret key could be recovered from the vault even when a fingerprint template and a query fingerprint are not identical. This scheme is considered as the best framework to date in addressing bio-cryptography (Boult, Schdrer & Woodworth 2007).

Clancy, Kiyavash & Lin (2003) proposed the first attempt to implement a fingerprint fuzzy vault. The genuine extracted minutia points plus a number of chaff points will construct a fuzzy vault. To address the ordering problem of minutia positions, a surrounded nearest-neighbour algorithm is used for the minutia matching. In this implementation Reed-Solomon error-correcting codes are used for error correction. The big issue with this approach is that without fingerprint registration or alignment, finding the corresponding minutia pair between the query fingerprint and the fingerprint template is impossible. Since it is not desirable to reveal any information about the stored template, this is problematic and challenging.

Kanade et al. & Uludag & Jain (2005; 2006) proposed a fingerprint fuzzy vault that employs orientation field (the angle of the associated ridge) as a helper data to assist alignment or registration. The authors acknowledge that the orientation field property does not reveal enough information to rebuild the fingerprint template. In this scheme a secret key *S* of length *16n* bits, where *n* is the degree of the encoding polynomial, is appended with an IBM Cyclic Redundancy Check (CRC) 16, for error-correction encoding, which is then called *SC*. Then *SC* is divided into non-overlapping 16-bit segments and each of them is used as a polynomial coefficient.

$$P(x) = C_n X_n + C_{n-1} X_n - 1 + \cdots C_0 \qquad (5)$$

The attributes of a minutiae point, namely *u, v* coordinates are represented as 8-bit strings. Concatenating the bit strings corresponding to *u* and *v* will result in a 16-bit code, noted as x. The polynomial P is calculated at all the points in the selected minutia set x to achieve the set $(x) = \{P(x_n)\}_{n=1}^{r}$, where r is the number of extracted minutia points. The corresponding elements of set x and $P(x)$ build a *locking set L*:

$$L = \{(x_n, P(x_n))\}_{n=1}^{r} \qquad (6)$$

Then a random chaff points set is also created.

$$C = [(c_1, d_1), \ldots, (c_m, d_m)], \text{ Where } d_i \neq P(c_i) \quad (7)$$

Finally, the randomly reordered combination of chaff set and locking set constructs the vault.

For decoding, the minutia points are retrieved from a given query fingerprint sample. If these points overlap considerably with the template's points, then it is possible to identify many points in the vault that lie on *P*. If an adequate number of points on *P* can be identified, it is feasible to reconstruct *P* and in doing so decode the secret key. If an adequate number of points on *P* cannot be identified, reconstructing *P* will not be feasible and the authentication fails. A sample fuzzy vault scheme procedure is illustrated in Figure 8. Based on the authors' acknowledgement, retaining the orientation field of the ridges significantly improves the effectiveness of the scheme and can prevent having to save the fingerprint template. However, it is uncertain how much the distortion would affect the results. Also, this scheme relies on accurate registration which is impossible due to the high probability of noise and errors. Uludag & Jain (2006) have made an attempt to address the registration problem by adopting line-based minutiae features suggested by Malickas & Vitkus (Malickas & Vitkus 1999); however a Receiver Operating Curves (ROC) performance showing the dependence of authentic acceptance rate from false acceptance rate has not been provided.

Later on, Nandakumar, Jain & Pankanti (2007) introduced a modified fuzzy vault scheme that employs orientation-field based helper data, called High Curvature Points, to help in alignment. In this approach, accuracy of the helper data extraction strongly influences recognition performance. Therefore, failure in extracting the helper data increases the False Rejection Ratio (FRR) which makes this approach unsuitable for practical applications.

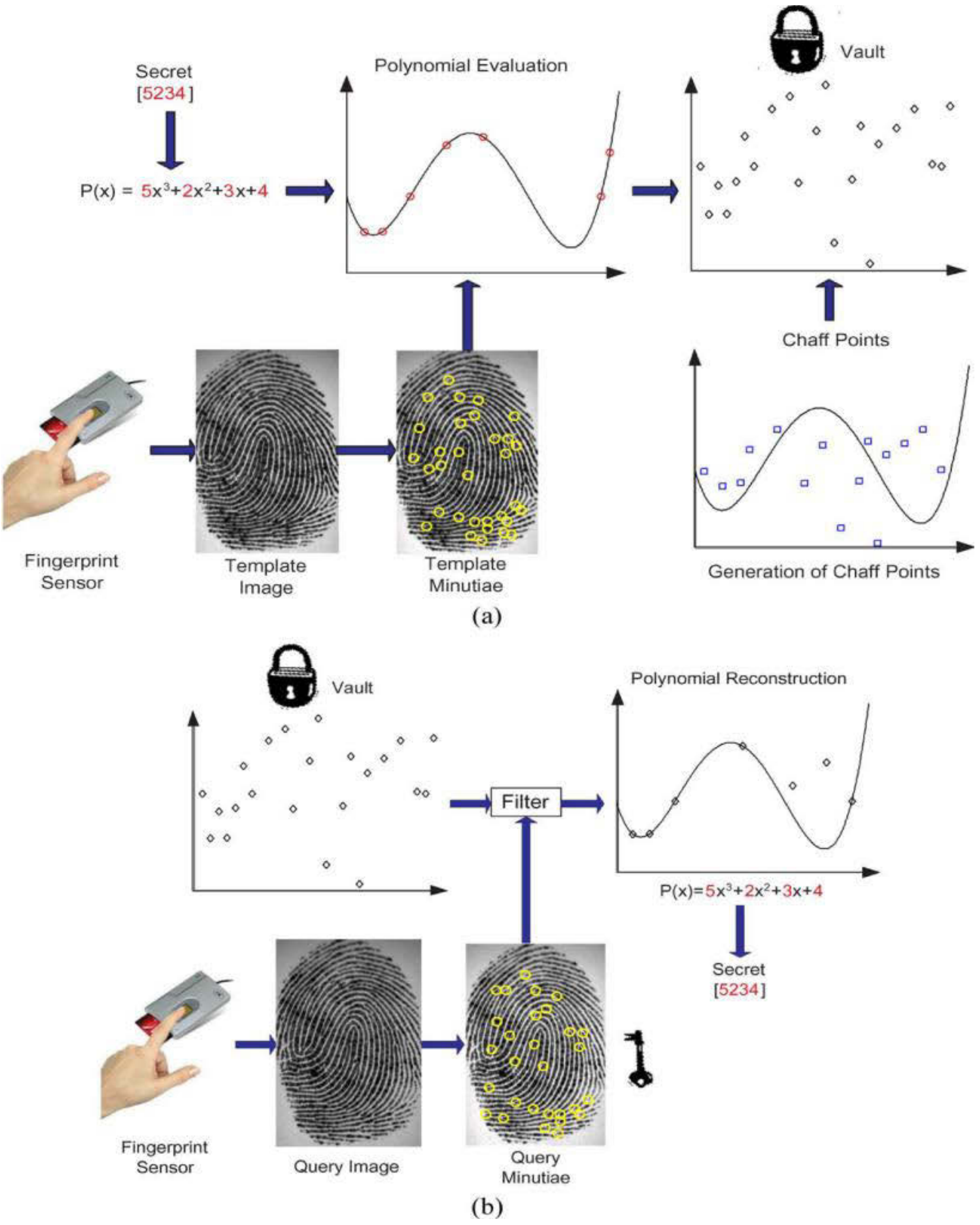


Figure 8: Fuzzy vault operation scheme. (a) Vault encoding. (b) Vault decoding (Nandakumar, Jain & Pankanti 2007)

Shenglin & Verbauwhede (2005) constructed an automatic secure fingerprint verification system based on the fuzzy vault scheme. The construction of the fuzzy vault during the enrolment phase is automated by aligning the most reliable reference points between different templates, based on which the converted features are used to form the locking set. The size of the fuzzy vault, the degree of the underlying polynomial, as well as the number of templates needed for reaching the reliable reference point were investigated.

Daesung et al. & Feng et al. (2008; 2005) proposed the first approach of automatic fingerprint alignment by using a geometric hashing technique which has been used for model-based object recognition applications to solve the fingerprint verification problem in vault decoding. Their approach consists of two phases; enrolment and verification. The enrolment process includes a minutia information acquisition stage which consists of minutia coordinates, angle and type (ending/bifurcation) and hash table generation. Let $m_i = (x_i, y_i, \theta_i, t_i)$ represent a minutia and $L= \{m_i \mid 1\leq i\leq r\}$ be a locking set including the genuine and chaff minutiae. In L, the genuine and chaff minutiae can be represented by $G = \{m_i \mid 1\leq i\leq n\}$ and $C= \{m_i \mid n+1\leq i\leq r\}$, respectively. In the hash table generation step, an enrolment table is generated in such a way that no alignment is needed in the verification process for unlocking the vault by using the geometric hashing technique. The details of the hash table generation can be found in (Daesung et al. 2008). In the verification process, direct evaluations without alignment are performed in 1:1 matching between the enrolment hash table and captured fingerprint input image, in order to select sufficient number of the genuine minutiae (G).

The geometric hashing-based fuzzy fingerprint vault solution can solve the auto alignment problem, yet it requires more memory space due to the large size of the hash table (Sungju, Daesung, Hanna, et al. 2008; Sungju, Daesung, Woo Yong, et al. 2008). Furthermore, it is proved that the vulnerability generated by the introduction of geometric hashing can reduce the security of the system dramatically and allow a potential cracking algorithm against it (Guo Xiao & Hu Ai 2009). Sungju, Daesung, Hanna, et al. and Sungju, Daesung, Woo Yong (2008; 2008) introduced an approach to reduce the static memory requirement by using the time-memory trade off. In this way, the hash table is generated on-the-fly at the verification phase instead of the enrolment phase. In addition, to reduce the dynamic memory requirement, the basis set which is used in hash table generation is selected carefully to minimize the possible degradation of the verification accuracy and the execution time. Guo Xiao & Hu Ai (2009) proposed a modified fuzzy fingerprint vault algorithm based on geometric hashing to solve the cracking algorithm drawback due to the characteristics of geometric hashing technique. The basic idea is to scatter the distribution of genuine minutiae in vault domain. The method increases the degree of the polynomial and results in a dramatic rise in the execution time which is not applicable for online applications.

AlTarawneh, Woo & Dlay (2008) made an attempt to use other fingerprint features vectors instead of using the minutiae to construct the fuzzy vault. A feature vector is the collection of both the global pattern of ridges and valleys and the local characteristics in each filtered image. They claim that their system has a high complexity for attackers and acceptable verification accuracy. They did not provide any measure of FAR (False Acceptance Rate), FRR (False Rejection Rate) and GAR (Genuine Acceptance Rate) to evaluate the verification accuracy.

Recently, Kai & Jiankun (2009) proposed a new pre-alignment free fuzzy vault scheme incorporating the Hierarchical Structure Check (HSC) minutia matching algorithm. They acknowledge using a translation and rotation invariant composite feature to replace minutiae location coordinates, which improves GAR.

Table I shows some of the key research efforts that have been carried out so far on the fingerprint fuzzy vault scheme in chronological order:

Table 7: Summary of Fingerprint-based Fuzzy Vault Implementation

Author	Features	Year
Jules & Wattenberg	Error-Correcting Codes	1999
Jules & Sudan	Theoretic Framework	2002
Clancy et al.	Nearest-Neighbor Algorithm	2003
Uludag et al.	Orientation Field	2005
Yang & Verbauwhede	Reference Minutiae Points	2005
Chung et al.	Geometric Hashing Technique	2005
Hu & Xi	Hierarchical Structure Check	2005
Nandakumar et. al	High Curvature Points	2007
Altarawneh et al.	Features Vector	2008
Kai & Jiankun	Minutiae Composite Feature	2009

4.1.3 Irrevocable Key from Cancellable Fingerprint Template

Fuzzy vault is a novel and promising solution to develop a cryptographic method to protect the fingerprint features in such a way that only legitimate user can access the secret data. Nevertheless, impostors can still compromise them by cross-matching if they could steal multiple vaults of a legitimate user (Scheirer & Boult 2007). Since a user has only ten fingers, thus if the fingerprint's data are compromised, the user may quickly run out of biometric data to be used for authentication and cannot re-enrol forever. A different approach for temporary fingerprint template protection is that of using a transformed fingerprint feature, called cancellable fingerprint. This scheme preserves the privacy since it is not possible or rather it is computationally very hard to recover the original fingerprint information using such a transformed version. In this method, it is possible to provide revocability since in case the fingerprint template is compromised, it can be re-enrolled using another transformation function. Another main advantage of this scheme is that it prevents cross-matching between applications since each application using the same fingerprint would use a different transformation (Fengling et al. 2007; Ratha et al. 2007).

Ratha et al. (2007) have formalized the challenges of using a cancellable template for fingerprints in four main categories as follow:

1- **Registration;** An accurate registration is needed to match minutiae pair,

2- **Intra-User Variability Tolerance;** the probability of FRR should not increase in the transformed domain,

3- **Entropy Retention;** the probability of FAR should not increase in the transformed domain,

4- **Transformation Function Design;** it should not leak any information about the fingerprint template.

They proposed three transform methods: Cartesian, Polar and surface. The detailed information about these methods can be found in (Ratha et al. 2007). Their approach also requires pre-alignment of fingerprint images and therefore is not robust to image registration.

Chikkerur et al. (2008) proposed a registration-free cancellable fingerprint construct based on localized, self-aligned texture features. They extracted an $N{\times}N$ pixel patch around each minutia instead of storing minutiae information. Based on their assumption each patch provides information about the unique identity of the individual.

Lalithamani & Soman (2009a, 2009b) made attempts to extract the minutia points that were altered, in an efficient manner, so as to acquire transformed points. Consequently, those points were used to produce the irrevocable templates which were in turn utilized for the extraction of irrevocable keys. They claimed that it is highly infeasible to acquire the cancellable fingerprint templates or the original fingerprint from the generated key, but they have not presented any experimental results on their claims.

Recently, Xu & Wang (2010) proposed a scheme for cancellable fingerprint fuzzy vault based on a chaotic sequence. They used transformed templates to construct vaults instead of the original ones. Also, by using a Hash function to encrypt the key, they enhanced the security of the secret key. Finally, by using a box matcher to match minutiae between transformed templates and transformed samples, they could overcome non-linear distortions (Lalithamani & Soman 2009b). Yet, their experiment results were based on the pre-alignment assumption for the template and query which is not realistic.

Cancellable approaches have the benefit of template revocability; however most of them fully depend on accurate registration which is the most difficult challenge in any fingerprint identification system. Therefore, it can be said that the approaches which are registration error tolerant or registration free are considered to be the best and most realistic approaches in applied and practical systems.

4.2 Summary

A fingerprint template is "fuzzy" (Yuan 2009) due to;

- Variations in impression conditions
- Ridge configuration
- Skin conditions
- Acquisition devices variability
- Non- cooperative attitude of a subject

Therefore, two fingerprint readings from the same finger are rarely identical, even though they are highly similar. This challenges genuine acceptance from the cryptosystem (authentication of a legitimate user). In addition, cryptography relies on uniformly distributed random strings for its secret keys and guarding the secrecy of the keys is one of the major challenges in practical cryptosystems.

Allegedly, the fuzzy fingerprint vault could be a prominent solution to protect the fingerprint template while ensuring sufficient genuine acceptance. It can address missing or fake minutiae effectively as it requires only enough genuine minutiae to unlock the secret key. However, it cannot solve the distortion problem and it relies heavily on accurate registration. Thus, it can be pointed out that the main challenging issue is aligning two fingerprints as the original template is not available during the authentication process. Therefore, there is a vital need to extract some specific information called helper data to make the matching process possible (Uludag & Jain 2006). At the same time, the storage of such helper data should not ease the breaking of the encryption. In later chapters, this thesis proposes a modification of the scheme presented by Nandakumar, Jain & Pankanti (2007) in order to achieve a more accurate helper data extraction and improve the recognition performance with the fuzzy vault.

CHAPTER 5 FUZZY VAULT DESIGN AND IMPLEMENTATION

In this study, the focus is on a specific implementation of fuzzy vault by Nandakumar, Jain & Pankanti (2007). A complete implementation of both coding/decoding phases and helper data extraction for such a system would typically be a teamwork effort. In this work we have implemented the coding phase for the secret key and vault generation which in corresponding parts is identical to the decoding process. Adding it with the helper data has had to be left out because of its complexity. However, as an original contribution of this thesis, we have proposed a novel method to extract the curves in fingerprint images and compared it with existing methods. Curves are a global feature of fingerprints and curve extraction as a function is a key part of the helper data block. This global feature has also a major role in fingerprint classification. This chapter will mainly be a detailed description of the coding and decoding steps and simulation results. A discussion on helper data will be addressed in the next chapter.

Imagine that a user wishes to cover a cryptographic key by means of his or her biometric template. This template appears as an unsorted set, X. The user picks a polynomial that encodes the secret key and then evaluates the polynomial on every component of X. The user then picks numerous random chaff points outside the polynomial P. The whole set of points both lying on P, i.e., authentic points, and those outside P, i.e., chaff points, make up the vault V. The chaff points will obscure the authentic points lying on P from a hacker. The points lying on P could encode the entire template X as well as the secret key thus hiding these points will keep the template and the secret key safe, simultaneously.

Supplying another biometric sample (query) will attempt to recover the secret key from the vault. Let the query be another unsorted set, X'. If X' overlaps considerably with X, then the user can identify many points in V that lie on P. If an adequate number of points on P can be identified, an error correction scheme will be applied to exactly reconstruct P and, in so doing, decode the secret key. If X' does not overlap considerably with X, reconstructing P will not be feasible and the authentication failed. This technique is called a fuzzy vault since the secret key could be recovered from the vault even when X and X' are not identical.

As described previously, the Fuzzy Vault has two main parts: vault encoding and vault decoding which are described hereafter

5.1 Fuzzy Vault Encoding

The security of fuzzy vault scheme is based on the infeasibility of the polynomial reconstruction problem by adopting an error-correction coding component. In addition, fuzzy scheme can effectively deal with the case of minutiae missing and/or a fake minutia, as it requires only enough genuine minutiae to unlock the secret. But this scheme is limited in addressing minutiae distortion problem and also relies on accurate registration (Hu 2008). So there is a need to address these issues to enhance FAR and FRR parameters which are standards for evaluating the convenience and security of an effective biometric authentication system.

Steps of fuzzy vault encoding are:

- Minutiae features extraction
- Minutiae Selection algorithm
- Encode minutiae points
- Generate enough chaff points
- Encode chaff points
- Create a polynomial encoding the secret key
- Generate the locking set
- Construct the vault
- Extract the helper data.

Figure 9 illustrates the block diagram of the proposed fuzzy vault encoding scheme by Nandakumar, Jain & Pankanti (2007). Our contribution would mainly improve upon "Helper Data Extraction" block.

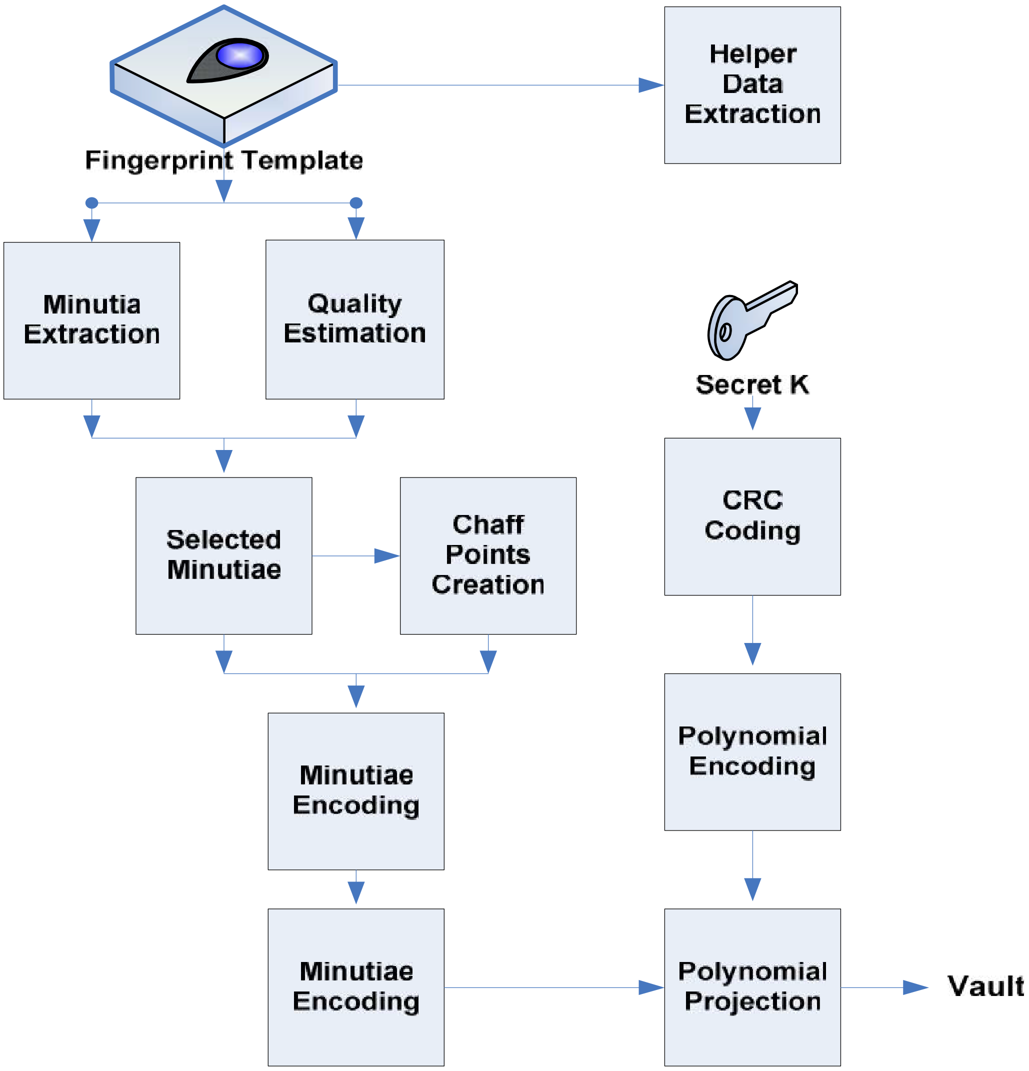


Figure 9: Functioning scheme of vault encoding

5.1.1 Minutiae Feature Extraction

Rao's algorithm described in (Jain, Lin & Bolle 1997) has been used for minutiae extraction in this project. The algorithm level design in this implementation is showed on Figure 10 as follows:

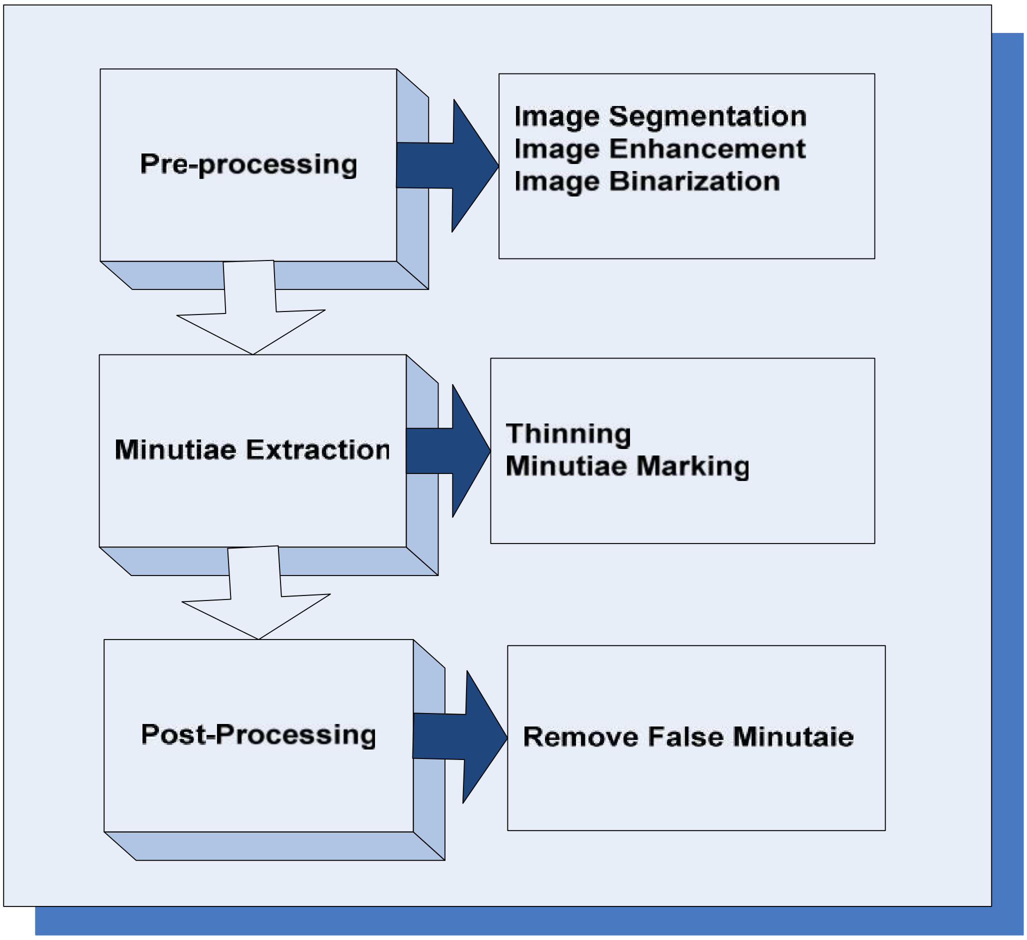


Figure 10: Algorithm Level Design

5.1.1.1 Pre-processing

The process of pre-processing is to obtain a binary segmented fingerprint ridge image from an input greyscale fingerprint image where the ridges are assigned the value *'1'* (white) and the rest of the image points are assigned the value *'0'*. This is achieved through the following steps which have been implemented in MATLAB:

- **Image Segmentation:** The first step of the fingerprint enhancement algorithm is image segmentation. When minutiae extraction algorithms are applied to the background regions of an image, it results in the extraction of noisy and false minutiae. Thus, segmentation is employed to discard these background regions, which facilitates the reliable extraction of minutiae.

  In this project combination of open and close Morphological Operations have been used for Region of Interest (ROI) extraction (Ratha, Chen & Jain 1995). The result of this implementation is shown in Figure 11. Mathematical Morphology is a tool for extracting image components that are useful for representation and description. The primary application of morphology occurs in binary images (Bovik 2005, pp. 135-56). The most basic morphological operations are dilation and erosion. Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels on object boundaries. In the morphological dilation and erosion operations, the state of any given pixel in the output image is determined by applying a rule to the corresponding pixel and its neighbours in the input image. A *morphological opening* consists of an Erosion followed by a Dilation; given that these operations are not inverse, the final result differs from the original image. A morphological opening changes every object pixel that is touching a background pixel into a background pixel. A *morphological closure* is instead a Dilation followed by an Erosion. This operation changes every background pixel that is touching an object pixel into an object pixel. Therefore, it can be said that an Erosion makes the objects smaller, and can break a single object into multiple objects. A Dilation makes the objects larger, and can merge multiple objects into one.

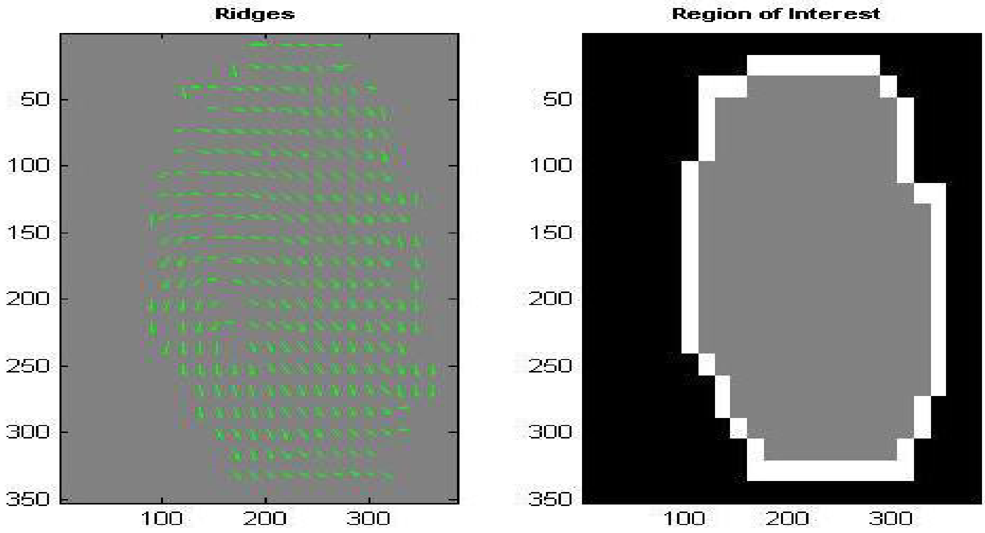


Figure 11: Region of Interest

- **Orientation Field Estimation:** This property is used for two distinct objectives. First and foremost to extract the helper data, in order to essentially enable adequately accurate alignment of the captured fingerprint with the registered fingerprint, and second, as an added characteristic to the minutiae coordinates for better differentiation and discrimination between minutiae, and therefore better detection.

  To decode the secret key, *K*, it is needed to match a fingerprint template with a fingerprint query. The first and most crucial problem in any fingerprint matching process is to align the fingerprint template and the query. In this implementation, helper data are extracted to enable this matching process in the vault decoding.

  The orientation field (Rao 1990) is used to compute the best possible central ridge direction in each 16*16 windows or block. The following steps are involved in the computation of the orientation field for each window (Ratha, Chen & Jain 1995). The results of these steps are shown in Figure 12.

  1- Compute the gradient of the smoothed block. Let G_x *(i, j)* and G_y *(i, j)* be the gradient magnitude in x and y directions, respectively, at pixel *(i, j)* obtained using a *3*3* Sobel mask.

2- Obtain the central direction in a *16*16* block using the following equation:

$$\theta_d = \frac{1}{2}\tan^{-1}\left(\frac{\sum_{i=1}^{16}\sum_{j=1}^{16} 2G_x(i,j)G_y(i,j)}{\sum_{i=1}^{16}\sum_{j=1}^{16}\left(G_x(i,j)^2 - G_y(i,j)^2\right)}\right), G_x, G_y \neq 0 \qquad (8)$$

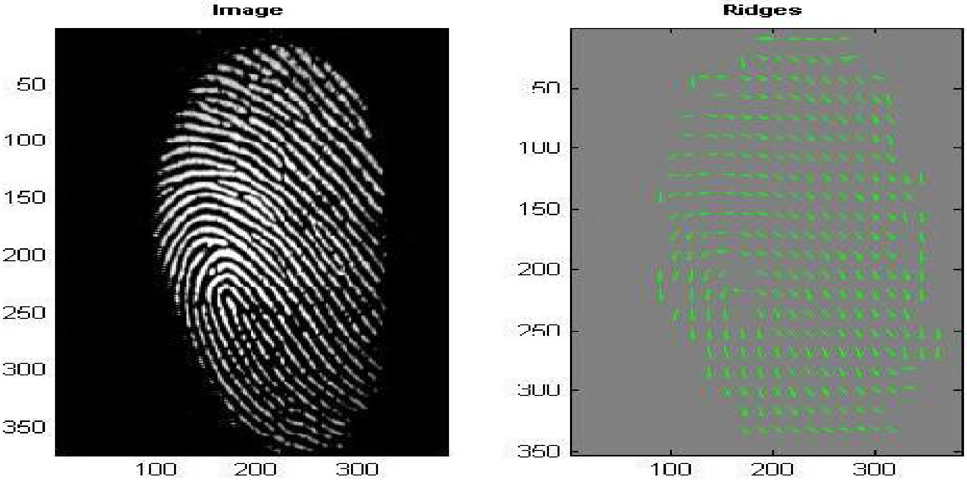


Figure 12: Orientation Field Estimation

- **Image Enhancement:** The following techniques have been used to give images better qualities for our purpose.

1- **<u>Histogram Equalization</u>:** This technique is helpful for images with backgrounds and foregrounds that are equally too bright or too dark. This technique typically improves the global contrast of the images, particularly when the actual data in the image is characterized by close contrast values. By this modification, the intensities will be well distributed on the histogram which means the regions with lower local contrast will acquire higher contrast.

Histogram equalization realizes this by effectively distributing the most frequent intensity values. Some advantages of this technique are straightforwardness and also being mostly a reversible operator. Moreover, the computation involved is not intensive. One drawback of this technique is that it is non-linear and its effects cannot be easily anticipated. It is possible that while decreasing the actual signal the contrast of the background noise is increased. However, by separating the foreground and background as we have done, this flaw is irrelevant (Acharya & Ray 2005, pp. 121-32)

2- **Fast Fourier Transform (FFT):** In this implementation, the FFT method (which is available in the MATLAB software) has been used for image enhancement. This method has been selected for use since the result after the FFT operation will have the same orientation. The enhanced image after Fast Fourier Transform has improved by connecting falsely broken points on ridges and also by removing spurious connections between ridges (Harne, Satao & Khan 2011).

Figure 13 shows a sample fingerprint image on the left and the same image after applying histogram equalization and FFT on the right.

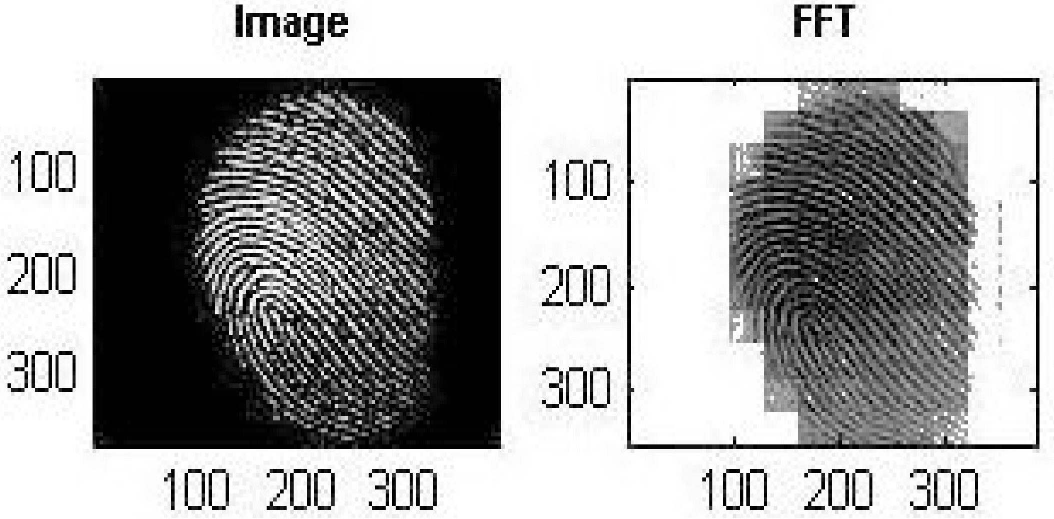


Figure 13: Fingerprint Image before (left) and after (right) Enhancement by Histogram Equalization and FFT

- **Image Binarization:** This process transforms 8-bit grey-level fingerprint image to 1-bit mage in which ridge pixels are represented by 1 and valley pixels by 0. After the operation, ridges in the fingerprint image as it is shown in Figure 14 are highlighted with white colour, while valleys are black.

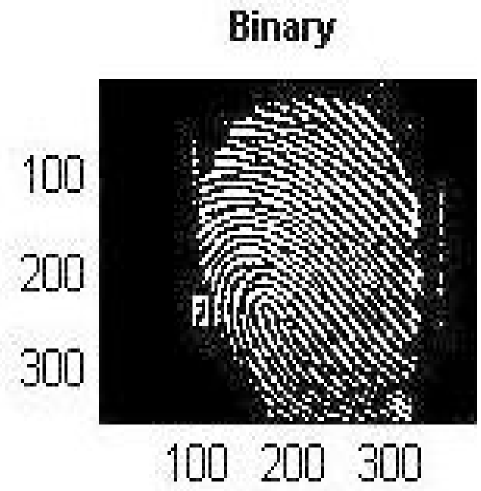


Figure 14: Binary Image

5.1.1.2 Minutiae Extraction

The process to extract minutiae proceeds through the following steps:

- **Thinning:** It is a morphological operation that successively erodes away the foreground pixels until they are one pixel wide. A standard thinning algorithm (Ratha, Chen & Jain 1995). is employed, which performs the thinning operation using two sub iterations. This algorithm is accessible in MATLAB via the *'thin'* operation under the *'bwmorph'* function. Each sub-iteration begins by examining the neighbourhood of each pixel in the binary image, and based on a particular set of pixel-deletion criteria, it checks whether the pixel can be deleted or not. These sub iterations continue until no more pixels can be deleted. Figure 15 shows a sample fingerprint before and after applying the thinning algorithm.

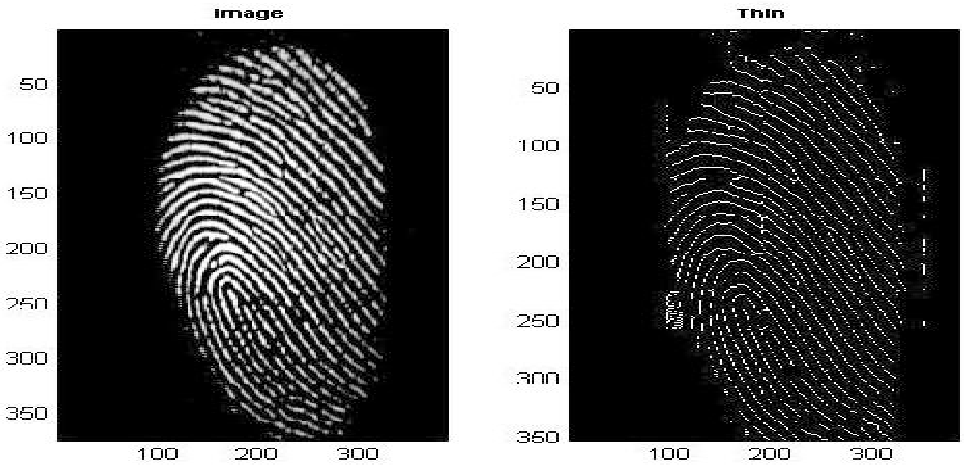


Figure 15: Image Thinning

- **Minutiae marking:** locating minutiae points in a thinned image is quite straightforward. In this implementation, the Crossing Number (CN) method is used to perform minutiae extraction. This method extracts the ridge endings and bifurcations from the image by examining the local neighbourhood of each ridge pixel using a 3*3 window. The *CN* for a ridge pixel *P* is given by (Alonso-Fernandez, Fierrez-Aguilar & Ortega-Garcia 2005):

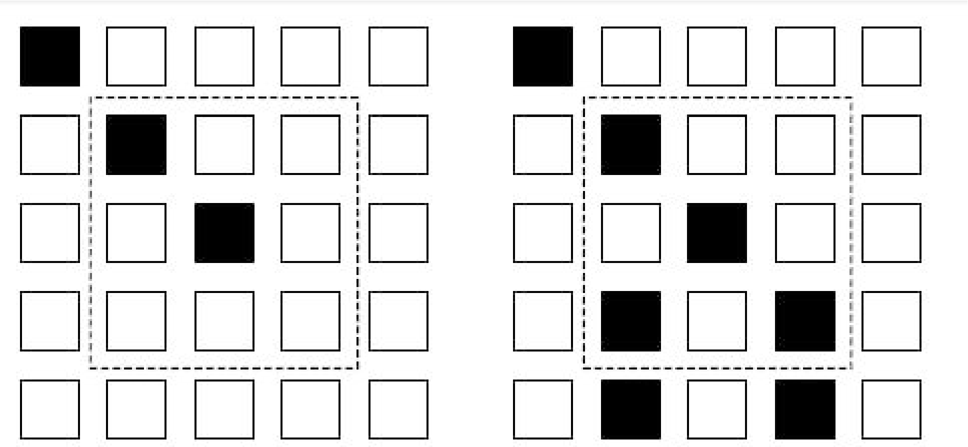
$$CN = 0.5 \sum_{i=1}^{8} |P_i - P_{i+1}| \quad P_9 = P_1 \tag{9}$$

where P_i is the pixel value in the neighbourhood of *P*. For a pixel *P* its eight neighbouring pixels are scanned in an anti-clockwise direction as depicted in Table 8.

Table 8: Pixel P and its Eight Indexed Neighbourhood Pixels

P$_4$	***P$_3$***	***P$_2$***
P$_5$	***P***	***P$_1$***
P$_6$	***P$_7$***	***P$_8$***

After the *CN* for a ridge pixel has been computed, the pixel can then be classified according to the property of its *CN* value. As shown in Figure 16 a ridge pixel with a *CN* of one corresponds to a ridge ending, and a *CN* of three corresponds to a bifurcation. In the other words, a pixel (x, y) is a ridge ending if $\left(\sum_{i=0}^{8} CN_i\right) = 1$ and a ridge bifurcation if $\left(\sum_{i=0}^{8} CN_i\right) > 2$ (Ratha, Chen & Jain 1995).



(a) *CN* = 1

(b) *CN* = 3

Figure 16: Examples of a ridge ending and bifurcation pixel. (a) A Crossing Number of one corresponds to a ridge ending pixel. (b) A Crossing Number of three corresponds to a bifurcation pixel.

In this implementation for each extracted minutiae point, the following information is recorded:

- x and y coordinates,
- orientation of the associated ridge segment
- type of minutiae (ridge ending or bifurcation)

Figure 17 shows an implemented result of a sample thinned image and the same image after locating all the minutiae points.

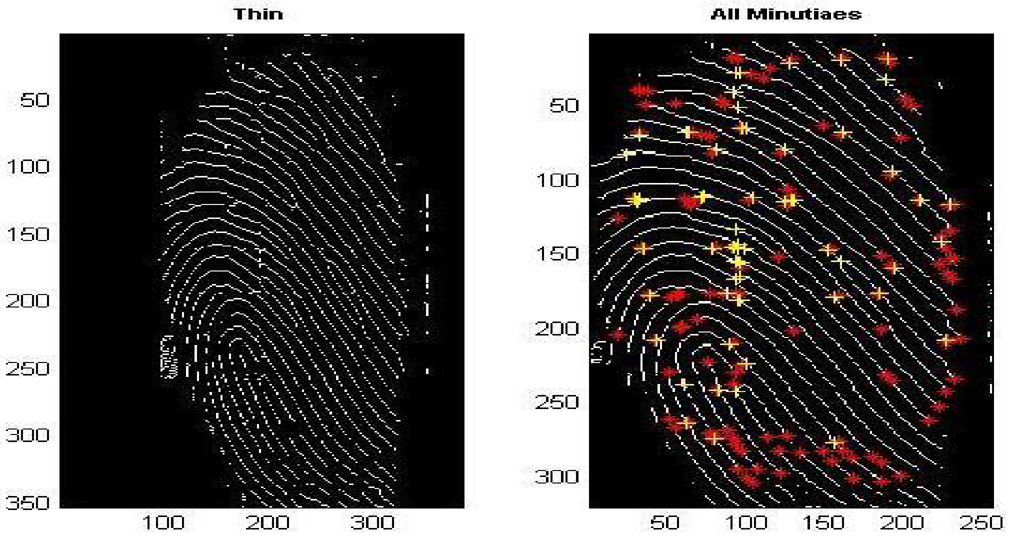
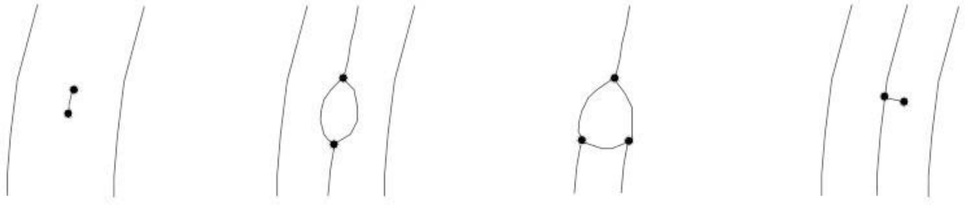


Figure 17: All Captured Minutiae Points

5.1.1.3 Minutiae Post-processing

All the ridge end points and ridge bifurcation points detected with this method are not always true features. False minutiae may be introduced into the image due to factors such as noisy images, and image artefacts created by the thinning process. Hence, after the minutiae are extracted, it is advisable to employ a post-processing stage in order to validate the minutiae. Figure 18 illustrates some examples of false minutiae structures, which include the spur, hole, triangle and spike structures (Wang et al. 2010). It can be seen that the spur structure generates false ridge endings; where both the hole and triangle structures generate false bifurcations. The spike structure creates a false bifurcation and a false ridge ending point.

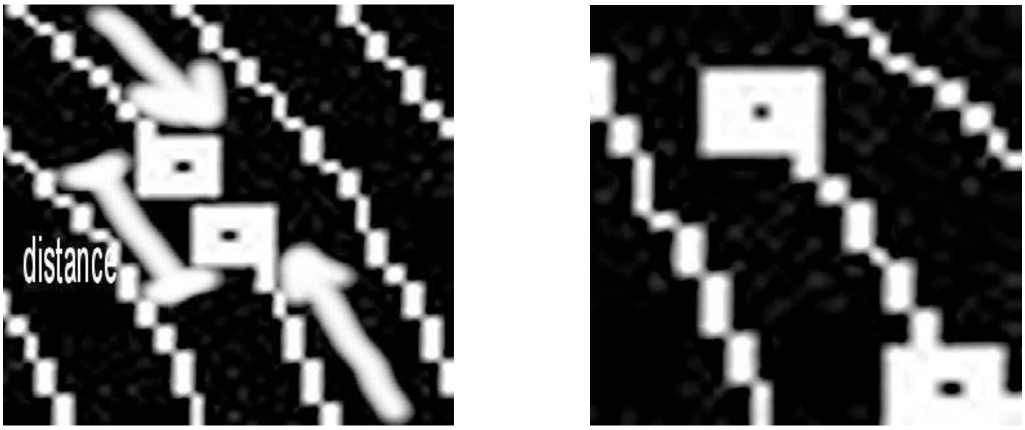


(a) Spur (b) Hole (c) Triangle (d) Spike

Figure 18: Examples of typical false minutiae structures

The majority of the proposed approaches for image post-processing in the literature are based on a series of structural rules used to eliminate spurious minutiae. One such approach is the one proposed by Ratha, Chen & Jain (1995) which performs the validation of minutiae based on a set of heuristic rules. For example, a ridge ending point that is connected to a bifurcation point, and is below a certain threshold distance is eliminated. This heuristic rule corresponds to removal of the spike structure shown in Figure 18(d). Additional heuristic rules are then used to eliminate other types of false minutiae. Furthermore, a boundary effect treatment is applied where the minutiae below a certain distance from the boundary of the foreground region are deleted.

Our approach in this implementation is removing two disconnected terminations short distance same/opposite direction flow or two terminations at a ridge which are too close. Figure 19 illustrates samples of short disconnected termination and also two close termination ridges. Figure 20 shows the result of removing the false minutiae.



(a) **(b)**

Figure 19: (a) Short disconnected termination (b) Two close termination

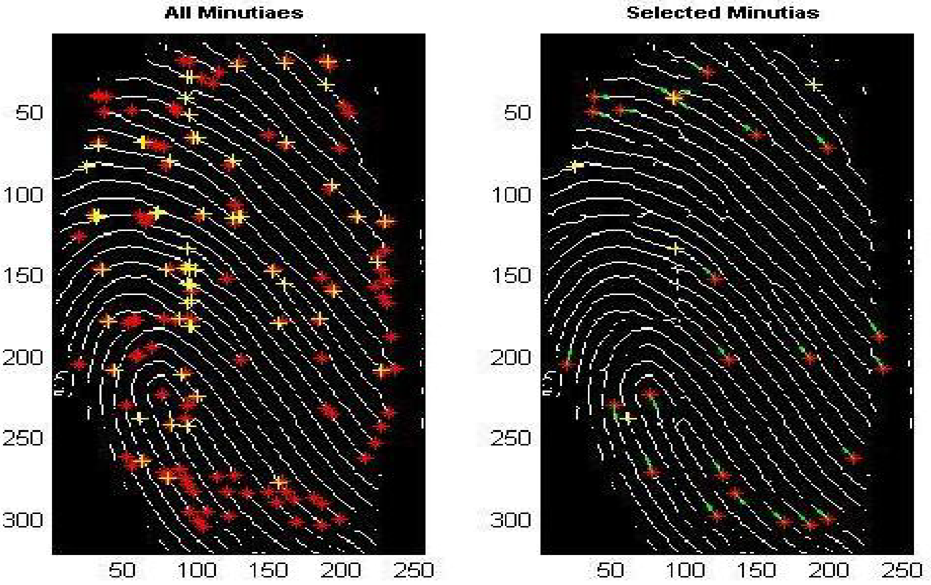


Figure 20: Minutiae points before and after removing the false minutiae

5.1.2 Minutiae Selection Algorithm

Since only ‘r’ genuine minutiae points are required to construct the vault, ‘r’ being the degree of the polynomial, and the number of extracted minutiae is usually much higher, it is possible to apply a minutiae selection algorithm. This selection algorithm sorts the minutiae based on their quality and sequentially selects the minutiae starting with the highest quality minutiae. Furthermore, the algorithm selects only well separated minutiae.

Evaluation of the quality index for each block can be done after distinguishing the fingerprint foreground from the background. For each foreground block B , let $g_s = \left(g_s^x, g_s^y\right)$ represent the gradient of the grey level intensity at site $s \in B$. The covariance matrix of the gradient vectors for a block of size $b*b$ in each site is given by (Kanade et al. 2005):

$$J = \frac{1}{b^2} \sum_{s \varepsilon B} g_s g_s^T \equiv \begin{bmatrix} j_{11} & j_{12} \\ j_{21} & j_{22} \end{bmatrix} \qquad (10)$$

So, the quality index for each block can be evaluated by:

$$k = \frac{(j_{11} - j_{22})^2 + 4 j_{12}^2}{(j_{11} + j_{22})^2} \qquad (11)$$

Figure 21 illustrates a sample of computing quality index for each block. The blocks with higher quality receive a brighter shade. The anomalies around the fingerprint will be ignored since there are no minutiae points associated with those blocks.

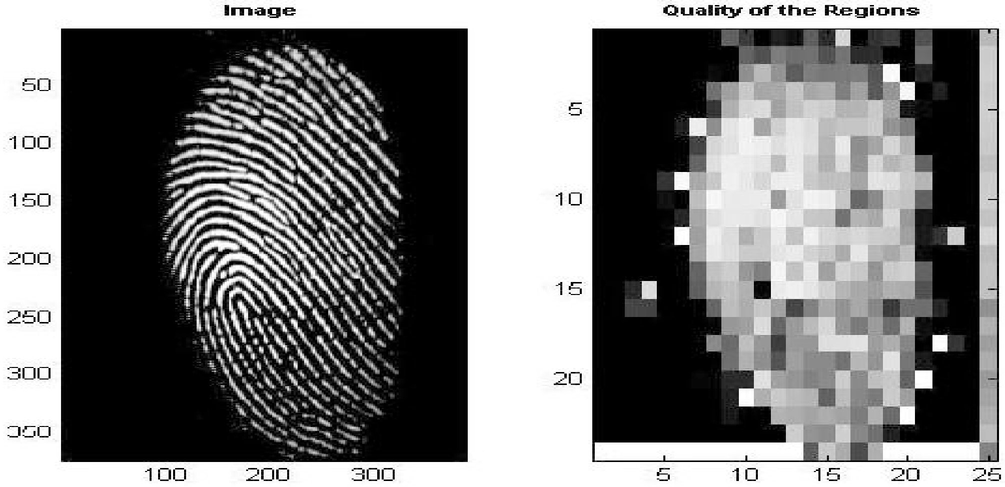


Figure 21: Quality Index Presented as the Brightness of Each Block

The algorithm selects just well-separated minutiae. We supposed based on our experiment on ten fingerprints in three different impressions that the minimum distance between any two selected minutiae points should be greater than a threshold, e.g., $\delta_1 = 20$. This selection ensures that minutiae are assigned unique values when they are encoded.

The distance D_M between two minutiae points m_i and m_j is defined as (Nandakumar, Jain & Pankanti 2007):

$$D_M(m_i, m_j) = \sqrt{(u_i - u_j)^2 + (v_i - v_j)^2} + \beta_M \Delta(\theta_i, \theta_j) \qquad (12)$$

where $\Delta(\theta_i,\theta_j)=\min(|\theta_i-\theta_j|,360-|\theta_i-\theta_j|)$ and β_M are the weight assigned to the orientation attribute which is set to 0.2 in our experiments.

Note that if the selection algorithm fails to find 'r' well separated minutiae, it is considered as a Failure-To-Capture (FTC) error and no further processing will take place.

5.1.3 Encode Minutiae Points

In mathematical domain, there exists a *finite field* with *p* elements for any prime number *p*. Finite fields are also known as Galois fields. The prime field is represented as *GF(p).* The prime field can also be created with p^m elements (an extension of *GF(p)* field) and it is represented by $GF(p^m)$, where *m* is a positive integer. By definition, arithmetic operations (addition, subtraction, multiplication, division, etc.) on elements of a finite field will always fall inside the field (Kotlarchyk, Pandya & Zhuang 2008).

Galois fields are frequently used for encoding and decoding digital block data. For digital data transmission and storage systems the $GF(2^m)$ fields are most commonly used. In our implementation the field $GF(2^{16})$ is chosen because it offers a sufficiently large universe (number of elements in the field) to ensure vault security and at the same time is computationally convenient for the fuzzy vault application (Nandakumar, Jain & Pankanti 2007).

After locating the well-separated minutiae points $\left(SM^T = \{m_j^T\}_{j=1}^{r}\right)$, they could then be encoded and assigned unique values in Galois fields $f = GF(2^{16})$. The attributes of a minutiae point, namely *u, v* and θ, are represented as bit strings of length B_u, B_v and B_θ, respectively. For each *u* and *v,* 5 bits have been allocated and θ is quantized and normalized within the 0-63 range to be represented by six binary bits such that B_u, B_v and B_θ add up to 16 bits. Therefore concatenating the bit strings corresponding to *u, v* and θ will result in a 16-bit code which can then be converted to a Galois field. Allocating six bits to θ means that the range of the Binary Coded Decimal (BCD) is 0-63, therefore the resulting resolution of θ is less than six degrees (360°/64 = 5.625°). Although this code length and bits arrangement appear to be satisfactory, note that the code length, the number of the allocated bits and also the arrangement of every single corresponding bit in the code design representing *u*, *v* and θ is arbitrary. A simple structure of the sixteen bits code and the allocated bits are shown in **Figure 22**.

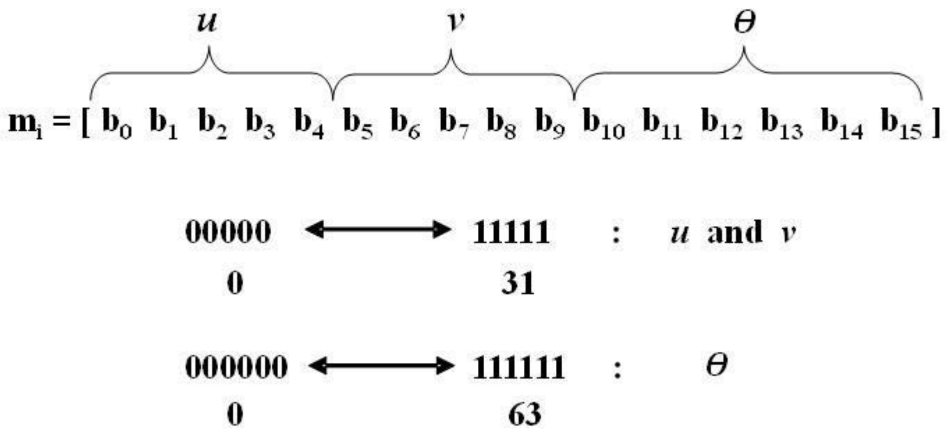

u
v
θ
mi = [b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 b10 b11 b12 b13 b14 b15]
00000
11111
:
u and v
0
31
000000
111111
:
θ
0
63

Figure 22: A simple arrangement of allocated bits in a sixteen bits code representing *v*, *u* and and their corresponding binary and BCD range.

5.1.4 Generate Chaff Points

The chaff point set $CM = \{m_k\}_{k=1}^{s}$ is created as follows. A chaff point *m*= (*u*, v, θ) is a randomly chosen point such that $u \in \{1,2,3, \dots, U\}$, $v \in \{1,2,3, \dots, V\}$ and $\theta \in \{0,2,3, \dots, 359\}$. The point *m* is added to *CM* if the minimum distance (as defined in equation 9) between *m* and all points in the set $SM^T \cup CM$ is greater than δ_1 .

In this implementation the number of chaff points is chosen to be ten times more than the number of minutiae points. It has been mentioned before that the number of chaff points influences the security of the vault, so a relatively large number of chaff points are added to increase the security.

5.1.5 Encode Chaff Points

After generating enough chaff points ($CM^T = \{m_k\}_{k=1}^{s}$), they could then be encoded and assigned unique values in Galois fields $f = GF(2^{16})$. In this step we do the same which we did to convert a minutiae point to Galois field.

5.1.6 Create a Polynomial Encoding the Secret Key

The fuzzy vault scheme is designed to secure a secret key, *K*, of length *16n* bits*,* where *n* is the degree of the encoding polynomial. Also, in order to increase the security, instead of a user selecting his/her own password (secret key (*K*)), the system would generate a random binary secret key of length 16*n bits. The system would then append a 16-bit CRC code to the secret key *K* to obtain a new secret key $K^{'}$ containing *16(n+1)* bits. The generator polynomial $G\ (w) = w^{16} + w^{15} + w^{2} + 1$ is commonly known as IBM CRC-16 is used for the CRC code. The secret key is encoded into a polynomial *P* of degree *n* (*n*=8 in our case) in a Galois field by partitioning *K'* into *n+1* quantities $c_0, c_1, c_2, \ldots c_n$, each constituted of 16-bits, and considering them as coefficients of the polynomial.

$$P(x) = c_n x^n + c_{n-1} x^{n-1} + \ldots + c_0 \tag{13}$$

Clearly, *P(x)* will be an element of the same Galois field as *x*.

5.1.7 Generate the Locking Set

In this stage, the polynomial P is calculated at all the points in the selected minutia set *x* to achieve the set *P* $=\{p(x_j)\}_{j=1}^{r}$. The paired corresponding elements of set x and *P* compose the locking set $L = \{(x_j, p(x_j))\}_{j=1}^{r}$.

5.1.8 Construct the Vault

Considering the chaff point set $CM = \{m_k\}_{k=1}^{s}$, a set $Z = \{z_k\}_{k=1}^{s}$ has been created by randomly generating values $z_k \in F$ such that the points (m_k, z_k) do not lie on the polynomial. Therefore the chaff set is defined as $c = \{(m_k, z_k)\}_{k=1}^{s}$. The combination of randomly reordered chaff set and locking set constructs the vault. So, the vault *V* is represented as $V = \{(a_i, b_j)\}_{i=1}^{t}$ where $t = r + s$. Just the vault *V* and the helper data set are stored in the system.

5.2 Implementation Results

This investigation is carried out in order to have a broader view of available technologies in the field which in turn will help us to find the missing links in the state of the art in this area.

In general the process of using any type of biometric authentication system requires several steps which include capturing of the raw biometric data, pre-processing and feature extraction, a recognition or matching algorithm and finally decision maker parts to accept or reject an individual. After the initial steps and creation of biometric template, careful consideration of the importance of biometric data and how it should be protected is required since biometric information typically is a very sensitive data and unique to every individual (Bala 2008).

To have a better overview, Figure 18 illustrates a sample fingerprint image and the resulting images after applying different algorithms at each step, including the original image, the image after Fast Fourier Transform (FFT), the binary image, the image after thinning, the ridges (i.e. to extract orientation field as an added dimension for the minutiae and also help to extract the flow curves for helper data), the regions of interest, all minutiae, selected minutiae after removing the spurious minutiae and the quality of regions. Note that a final set of minutiae will be picked based on a condition set for the quality of their regions to make sure that the best quality minutiae are selected. Selecting false minutiae could increase FRR.

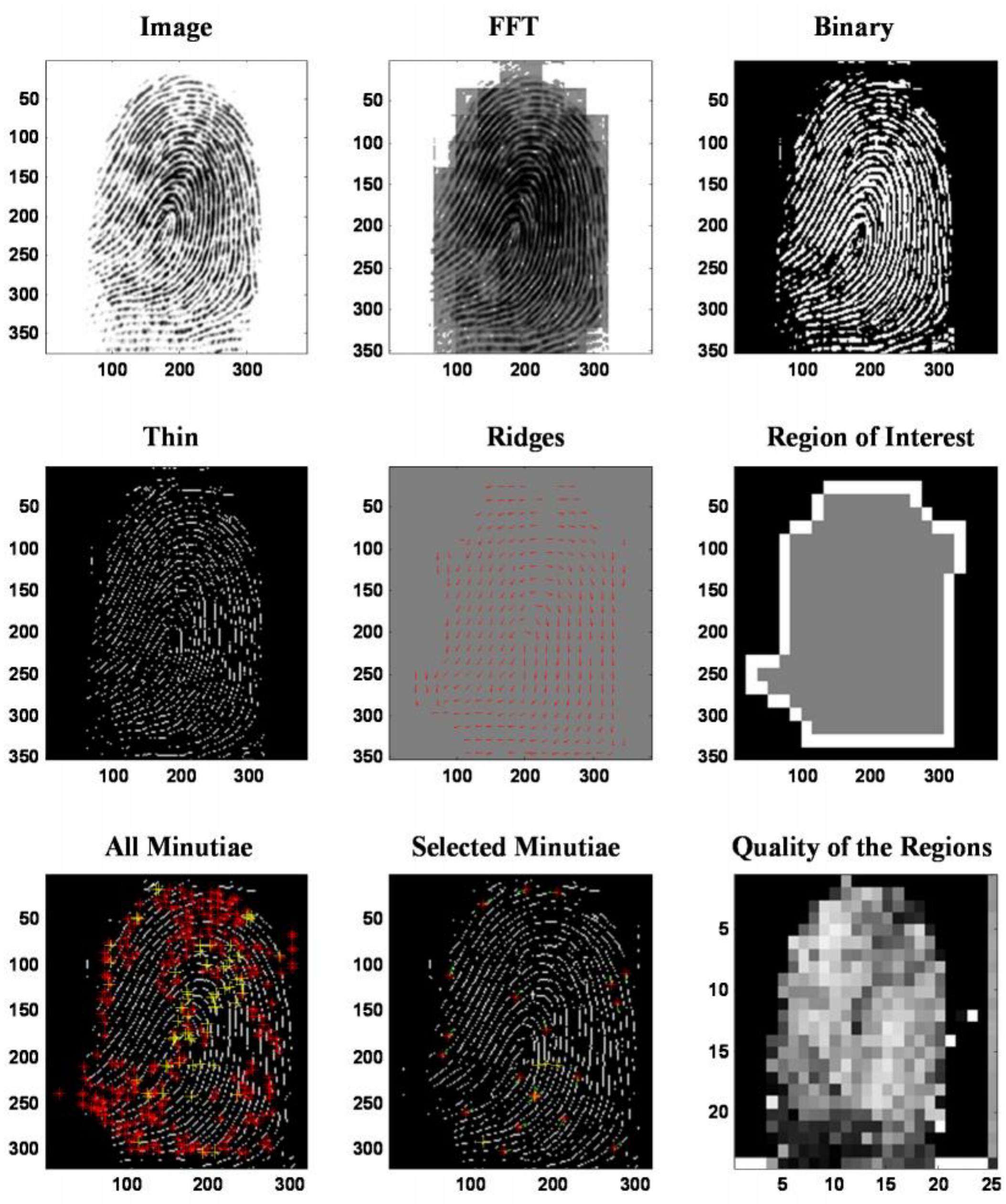


Figure 23: Minutiae Feature Extraction at a Glance

Cryptography is the practice and study of hiding critical information. Furthermore, cryptography not only protects data from theft or alteration, but can also be used for user authentication. But cryptographic algorithms suffer from the key management problem as they fully depend on the assumption that the keys will be kept in absolute secrecy. The fuzzy vault construct is a biometric cryptosystem that secures both the secret key and the biometric template by binding them within a cryptographic framework (Nandakumar, Jain & Pankanti 2007).

Figure 24 depicts a typical polynomial created in Galois field. This function contains a combination of both the fingerprint features and the secret key. Not only this function has to be extracted from a set which is a mixture of chaff points and genuine points (i.e. commonly ten times more chaff points than genuine points) but also it can be seen that even if this function is compromised it will not automatically reveal any information about fingerprint template including the location of the minutiae. This is the primary reason to generate this polynomial in a close field. Moreover, operations in Galois fields are very straightforward and computationally light.

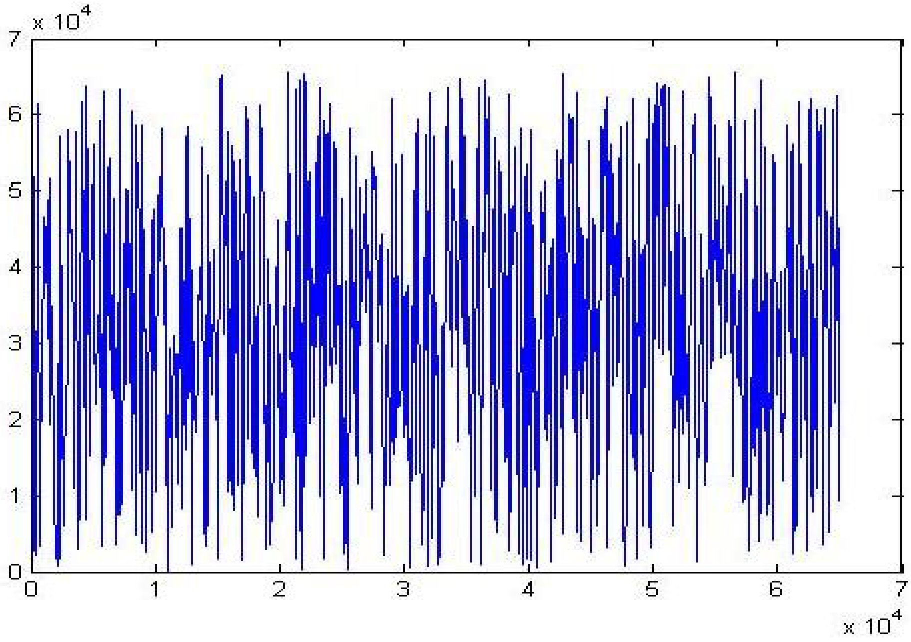


Figure 24: Created Polynomial in Galois Field

5.3 Summary

In this work, the state of the art in the automatic fingerprint authentication methods is thoroughly investigated and evaluated. The Fuzzy Vault authentication method appears to be the most promising of the proposed approaches. The implementation of the coding procedure for the Fuzzy Vault method was also achieved.

Generally, image processing, by its nature, is computationally expensive. Processing a captured fingerprint, like almost any other image processing, could greatly benefit from any improvement to the procedures which could reduce the computational load, hence reduce the computation time. Our main contribution in this work is to propose a novel method, which would reduce the computational load for extracting the curvatures from fingerprints which would be beneficial for processing helper data in both coding and decoding phases. This novel method could also be advantageous in classification of the fingerprints or any other fingerprint related process that relies on curvature extraction. The proposed method and the simulation results are presented in the upcoming chapters.

CHAPTER 6 A NOVEL APPROACH FOR CURVATURE DETECTION DESIGNED FOR HELPER DATA

6.1 Fuzzy vault helper data

The first and most crucial problem in any fingerprint matching process is to align the fingerprint template and query. Helper data are extracted to enable this matching process in vault decoding and are saved as public information in a fuzzy vault system. Therefore, while helper data should not disclose enough information to enable the reconstruction of a template, they should contain essential information to aid the matching process (Nandakumar, Jain & Pankanti 2007). Based on these prerequisites, it is well accepted to use global features of a fingerprint for helper data which are also commonly used for fingerprint classification (Dass & Jain 2004b).

The proposed approach in this chapter will improve upon current methods in computational time without losing precision of the extracted curve flows. The degree of improvement depends on the type of the fingerprint and its curves.

6.1.1 Global Fingerprint Features Extraction

Based on the definition of local and global fingerprint features, the first step in feature extraction is ridge detection. Fingerprint images can be characterized as oriented texture patterns. The direction of flow of the ridge structures at each location in the image can be characterized as a two-dimensional vector with unit norm. The orientation field (also called directional field) is defined as the set of orientation vectors for all sites in the image (Rao 1990). The orientation field at (x, y) is the angle ⍰$_{xy}$ of the fingerprint ridges, which cross through an arbitrary small neighbourhood centred at (x, y), form with the horizontal axis (Figure 2) (Tistarelli, Bigun & Grosso 2005).

A number of different approaches have been proposed for a reliable and fast estimation of the orientation field. These approaches consist of methods based on gradient-based approaches (Lin, Yifei & Jain 1998; Rao 1990; Ratha, Chen & Jain 1995), filter-based approaches (O'Gorman & Nickerson 1989) and Markov random field models (Dass 2004; Zhang, Brady & Smith 2001).

Rao's (1990) algorithm is the most commonly used method to compute the orientation field for minutiae (local features) extraction in fingerprint verification (Dass 2004). However, the Markov random fields for orientation field estimation proposed by Dass (2004) is specially designed for singularities (i.e. global features) detection in fingerprint classification (Ratha, Chen & Jain 1995).

6.1.2 Proposed Method

In general, at the global level, ridges often assume distinctive shapes characterized by high curvature and frequent ridge terminations (Maltoni et al. 2003). In order to extract a curve using a direction field, regardless of how the direction field is worked out, current methods use a step by step advancement from a starting point with fixed step size. If the step size is large the resolution and precision of the curve will be lost. If the steps are chosen too small, the computation cost will be high, which would be very important in any image processing procedure, especially in fingerprint matching application that large number of fingerprints may be needed to be examined. Taking a variable step size curve extraction approach will keep the integrity of the extracted curve while keeping the computation cost at minimum. The more visible differences will be at high curvatures.

6.2 Curve Extraction Implementation

For extracting curves on a fingerprint image the following steps have been implemented in MATLAB.

6.2.1 Block Orientation Field Estimation

The orientation image is hardly ever computed at full-resolution. Instead each non-overlapping block of size W × W, of the image is allocated a single orientation that correspond to the most principal orientation of each block. In this implementation, the block size is 16. Based on Rao’s algorithm, using a simple 3 × 3 Sobel mask gradient operator, the horizontal and vertical gradients, $G_x(x, y)$ and $G_y(x, y)$, for each pixel (x, y) are computed. The ridge orientation of each block is then evaluated by averaging the squared gradients within a 16 × 16 window (Rao 1990) (Equation 1):

$$\theta = \frac{1}{2} \tan^{-1} \left(\frac{\sum_{i=1}^{16} \sum_{j=1}^{16} 2G_x(i,j)G_y(i,j)}{\sum_{i=1}^{16} \sum_{j=1}^{16} 2G_x(i,j)^2 G_y(i,j)^2} \right) \tag{14}$$

A typical fingerprint and its orientation field are illustrated in Figure 25.

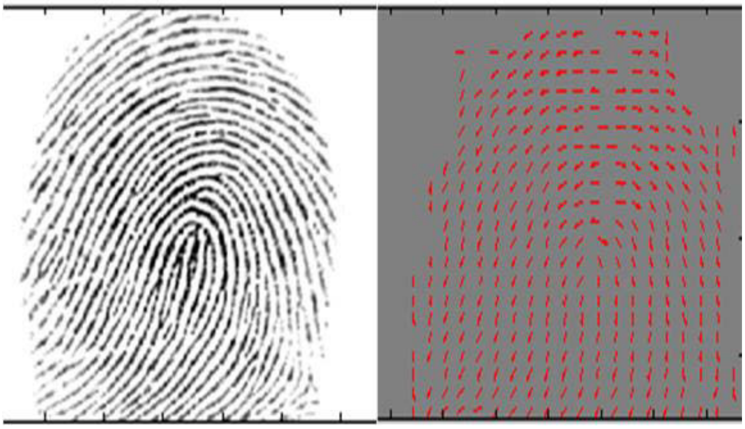


Figure 25: A sample fingerprint and its orientation field

6.2.2 Estimating the Orientation in each Pixel

In order to determine the curves within a fingerprint the orientation at any point may be required. Having the orientation of each block and its surrounding blocks the orientation in each pixel of that block could be estimated. The orientation of each pixel in a block is a function of the orientation of that block and the surrounding blocks, based on the position of the pixel in the block. Four blocks will affect the orientation at any point p=(x, y), which include the block that encloses the point and the three closest blocks to the point.

If the block size is B (16 in our case) then the enclosing block of point p will be the block index [m, n], where m=ceil(x/B) and n=ceil (y/B). The function “ceil”, already a function in MATLAB, is a round up function and is where ceil(z) is the smallest integer number larger than z. For example, ceil(4.01)= ceil(4.99)=5. Then we define:

$$\begin{cases} r = \mathrm{sign}(x - m \square B + B/2) \\ c = \mathrm{sign}(y - n \square B + B/2) \end{cases} \tag{15}$$

where sign(z) is -1 if z is negative and is +1 if z is positive hence r (row) and c (column) will point to the closest blocks to point p. Therefore blocks that will affect the direction in point p are [m, n], [m+r, n], [m, n+c] and [m+r, n+c] blocks. Also we define:

$$\begin{cases} W_{m+r} = (x - m \square B + B/2) \square r + 0.5 \\ W_{n+c} = (y - n \square B + B/2) * c + 0.5 \\ W_m = B - W_{m+r} \\ W_n = B - W_{n+c} \end{cases} \tag{16}$$

where W_{m+r} and W_{n+c} are the distances of the point p from the horizontal and vertical medians of its block (the lines that pass through the middle of the block) and they will be used to assign weights to the influence of surrounding blocks. There is a 0.5 added to compute the distance to the exact middle of the block assuming that B is an even number. Note that $W_{m+r} + W_m = W_{n+c} + W_n = B$. The closer the p is to the middle of the block, the smaller W_{m+r} and therefore the bigger the W_m will be, which will then be used as a mean of giving more weight to the block. The orientation at point p can now be calculated through the following equation:

$$\theta_p = \frac{1}{2} \tan^{-1} \left(\frac{\sum_{\substack{i \in (0,r) \\ j \in (0,c)}} W_{m+i} W_{n+j} \sin 2\theta_{[m+i,n+j]}}{\sum_{\substack{i \in (0,r) \\ j \in (0,c)}} W_{m+i} W_{n+j} \cos 2\theta_{[m+i,n+j]}} \right) \tag{17}$$

where $\square_p$ is the orientation angle at point p and $\square_{[m,n]}$ is the orientation angle at block [m, n]. The weight factors W_z will be normalized automatically since they are repeated in both numerator and denominator. Now having the orientation at each point the curves could be extracted.

6.2.3 Curve Extraction

For curve extraction the typical approach consist of the following steps:

1. Take an arbitrary point k;
2. Find θ_{k_0};
3. From k_0 move in the direction of θ_{k_0} with step size of SL to reach point k_1;
4. Repeat the same steps from point k_1 to the next point;
5. Stop if you cross the edges of the fingerprint or after certain number of steps;
6. Repeat the same steps from *k* in the opposite direction;
7. The set of these points will make up one curve within a fingerprint, thus the same procedure needs to be repeated from other starting points within the fingerprint in order to obtain enough curves to serve the purpose;

In current methods, step length SL is an arbitrary fixed value. If SL is too large the extracted curvature will not follow the real contours of the fingerprints ridges and will lose its accuracy. If the SL is chosen to be too small the computation cost will increase. Our approach consist of a variable SL which it length will vary between a maximum and minimum length. SL will be adjusted according to the sharpness of the curvature.

In the new approach at every step, the initial SL at every step is simply the maximum SL. Then the orientation angle at the new site will be calculated. The difference between the previous orientation angle and the new orientation angle (|⍰⍰|) will be used to adjust the SL accordingly using a simple linear relation shown in fig. 4. At $|⍰⍰|_{\mathrm{min}}$ the SL will be at its maximum length and vice versa. In our implementation SLmin=2 and SLmax=16.

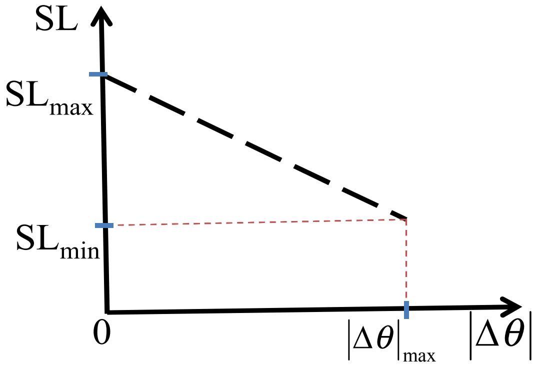


Figure 26: Linear relation between SL and □θ

Although other nonlinear relations could also be adopted but the objective is to keep the computation at minimum and this simple relation deems adequate. Note that many curves in fingerprints are soft curves and choosing SL to be small enough to capture the sharp curves will greatly increase the overall computation cost for extracting them.

6.3 Experiment Results

The quality of the extracted curves using large SL, small SL and variable SL (our method) is illustrated on a sample fingerprint in Fig. 5 (a, b, c, d).

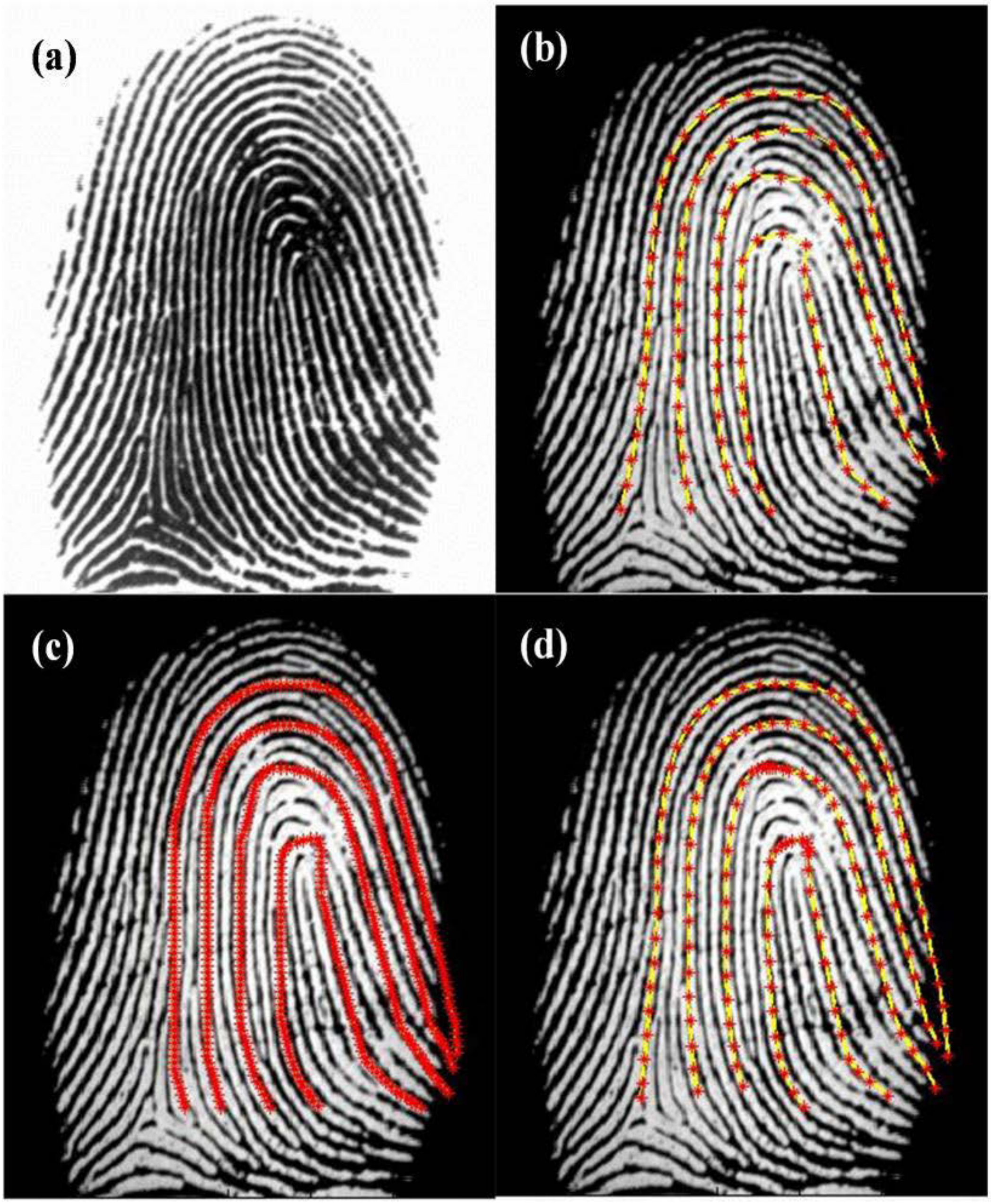


Figure 27: (a) Original Fingerprint (b) Extracted Curves with large SL (c) Extracted Curves with small SL (d) Extracted Curves with variable SL

As logically would be expected, the curves are more accurately extracted when SL is small (Fig. 5-c) compared with the ones with large SL (Fig. 5-b). Although, the number of points, hence the amount of calculations, with small SL, i.e. 2, in Fig. 5-c is eight times the ones with large SL, i.e. 16, in Fig. 5-b.

However, simply by regulating SL, based on the curvature of the contours (Fig. 5-d), the results are quite as fine as using very small SL (Fig. 5-c). Yet, for the soft curves, the computation cost with using variable SL is the same as using large SL. Compared to using small SL, the total computation cost in variable SL case is increased slightly, only at sharp curves, to capture the true contours of the ridges of the fingerprint. However compared to using small SL, using variable SL could considerably reduce the computation cost close to the order of SLmax/SLmin. In essence, variable SL will only reduce the SL where it is required and will keep it large where the curve is soft.

6.4 Variable Step Method Performance Evaluation

Amongst others, image processing deals with qualitative aspects of images e.g. image enhancement. Quantifying a qualitative feature, for example the sharpness, of a picture is not necessarily a straightforward task. If image processing involves a pattern recognition scheme, quantifying the quality of the pattern recognition would still be challenging. Extracting curvature lines from a fingerprint involves both image processing and pattern recognition aspects.

A novel method for extracting curves from a fingerprint was offered in the previous sections. Principally, we have argued that the new method is advantageous over using a fixed step scheme. By using this method, the fidelity of the extracted curves is preserved, which is one major objective, and at the same time the computational load is optimized, which is another significant objective. Let us assume three scenarios for choosing the step sizes and compare them with each other concerning their computational load. One scenario would be using the maximum step length, which we assume to be at most equal to one block side, in this case 16. Another case would be using the shortest step size which will be equal to one pixel and finally using the proposed method of variable steps. For simplicity let's refer to using the maximum step size as "LS" (Large Step method), using the variable step size as "VS" (Variable Step method), and using the small step size as "SS" (Small Step method).

At this point, let's use the analogy of driving a car in the course of a winding road. If we take snapshots of the car in equal time intervals, a pattern similar to Figure 28 should appear.

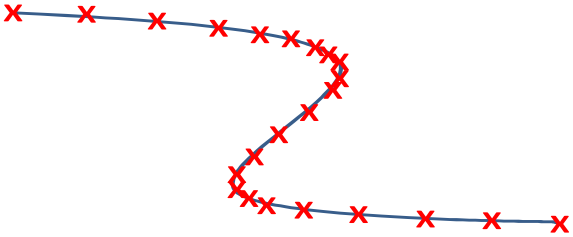


Figure 28: Snap shots of a car's position driving through a winding road.

Realistically, assuming a reasonably good driver, the expectation is for the driver to go faster where the road is straight or less curved and to slow down wherever the road bends, in order to better control the car. Driving with a constant, low speed (analogous to SS) throughout the trip will suffer from additional time cost and driving fast throughout the trip (analogous to LS) will have the danger of sliding off the road where the road is bending. The obvious choice would be to drive with variable speed (analogous to VS) depending on the curvature of the road.

One could reasonably estimate that times that will be incurred by using LS, VS and SS (namely t_{LS}, t_{VS} and t_{SS}) for the same curve, in the majority of cases, will be in the following relationship:

$$t_{LS} \leq t_{VS} \leq t_{SS} \tag{18}$$

In special cases the inequality may reverse. For example if the line has the maximum curvature throughout, the VS method will take the shortest step all the way which would computationally be equal to SS plus an extra load of step size calculation. In this special case $t_{SS} < t_{VS}$. Also the relationship between t_{LS} and t_{VS} could conceivably but infrequently reversed in cases where the LS method drastically skids off the curve and changes course into another groove making the curve much longer. Of course in such a case t_{LS} will be larger than t_{VS} only because it follows a totally false curvature.

Although physical speed is only a metaphor in our scenario, we draw on the above analogy to justify our choice for a variable-step method. Such a method is expected to attain comparable fidelity to a short-step method and computational times not much in excess of the long-step method. Note that the process of extracting curves in one fingerprint has to handle many individual curves across the surface of the fingerprint. Examining different classes of fingerprints in Figure 29, one could see that the expected improvement in time using VS versus SS could vary significantly for individual fingerprints as a consequence of variations in the curvatures of each fingerprint. However the objective is to investigate the average overall saving in time to perform VS versus SS which would be an indication of overall computational load improvement. Also the average additional time needed to perform VS versus LS will be presented. This would indicate how much additional computational load VS will impose compared to LS. Note that LS is primarily not suitable for extracting the curves from fingerprints because of lack of fidelity.

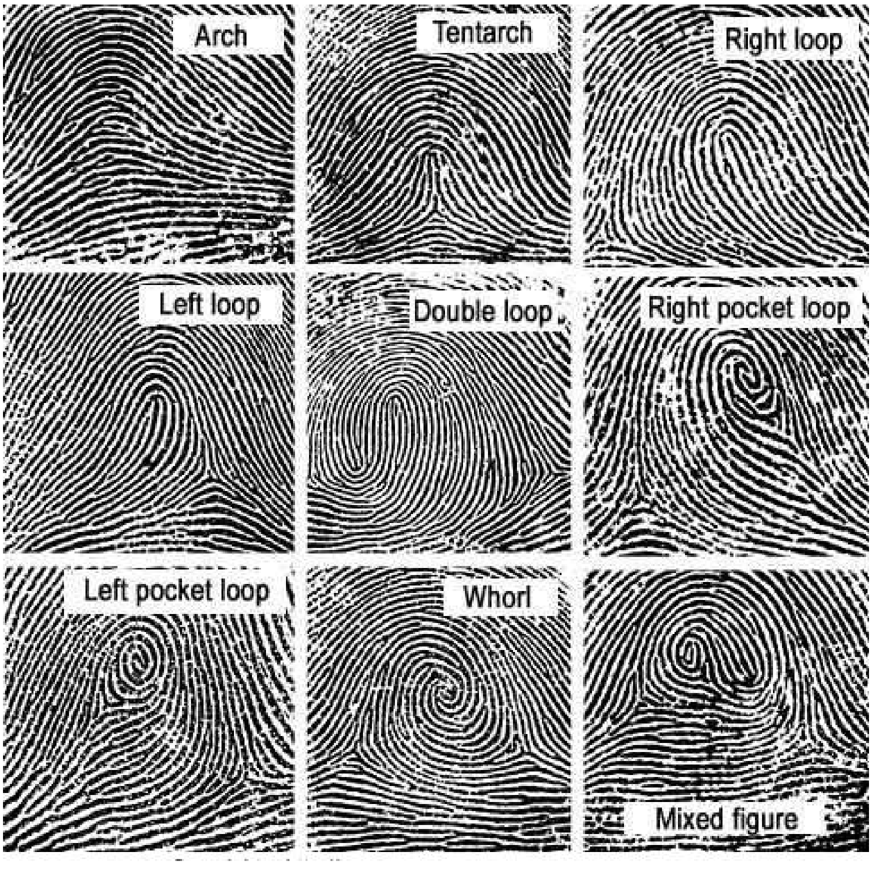


Figure 29: Classes of Fingerprints

Mainly, the difference between the time that it takes to perform LS, VS and SS corresponds to the difference in their computational load. The absolute value of the time itself is not of any concern since it will depend on the arch (curvature), length and number of the curve(s) extracted (that is dependent on the type of the fingerprint) and also the specifications of the system running the program, e.g. its processor, memory, operating system, etc.

As a result of the lack of fidelity of LS method t_{LS} is not always an accurate indicator of the minimum time for extracting every single curve since, in some cases; it may not even continue on extracting the correct curve. Nevertheless LS is accurate enough to be loosely used as a point of reference. t_{LS} is merely implied as the minimum time for fixed step size method for practical purposes.

We define three benchmarks based on the ratios of the execution times of LS, VS and SS methods as follows:

1. $\frac{t_{SS}}{t_{VS}}$: This benchmark will indicate the relative additional time required to perform SS compared to VS. SS is a viable method therefore any improvement in comparison to SS would be advantageous. This benchmark would be the most important indicator of VS's performance and is desired to be as large as possible (i.e. as close as possible to $\frac{t_{SS}}{t_{LS}}$).
2. $\frac{t_{LS}}{t_{VS}}$: Presuming that t_{LS} is the minimum time to perform curve extraction, $\frac{t_{LS}}{t_{VS}}$ would indicate the relative time saving when performing LS rather than VS. Looking at it in another way, it indicates the extra effort that it takes to perform VS versus LS $(1 - \frac{t_{LS}}{t_{VS}})$. This benchmark is favoured to be closer to one.

3. $\frac{t_{SS}}{t_{LS}}$: This ratio hypothetically shows the ratio between the maximum and minimum possible times to perform curve extraction. $\frac{t_{SS}}{t_{VS}}$ is desired to be as close as possible to this ratio.

The ratio of $\frac{t_{SS}}{t_{VS}}$ is more reliable and meaningful than $\frac{t_{LS}}{t_{VS}}$ and $\frac{t_{SS}}{t_{LS}}$ since the LS method may, as indicated before, in numerous cases, stray away from the original curve and lead to extracting false curves. Also, at first glance, it may seem that $\frac{t_{SS}}{t_{LS}}$ should be close to 16 which is the ratio between the step sizes in LS (which is 16) and SS (which is 1) but as will be seen the result is quite different. The main reason is that, besides the computations required at each step on the curve, there are other parts of the program which will roughly add equal times to the numerator and denominator. Hence instead of $\frac{t_{LS}}{t_{VS}}$ the result will be something like $\frac{T+t_{LS}}{T+t_{VS}}$. Also, the difference between the curves yielded by the two methods could alter the expected result.

Ultimately, when assessing the VS method, larger $\frac{t_{SS}}{t_{VS}}$ (on average) would indicate a better performance for VS. Principally it would mean a larger saving in computation and time compared to the SS method. For example if $\frac{t_{SS}}{t_{VS}}$ on average is equal to two it means that the SS method on average takes twice as long as the VS method to be executed. Therefore, the VS method would, in this case, cut the execution time to half which means a 50% saving in time.

In order to carry out the measurements the following approach is adopted:

1. Libraries:
    a. The measurement is performed as a batch process on four libraries of fingerprints.
    b. Each library contains 80 fingerprints.

2. Computer setup:
    a. Computers nowadays are multitasking and switch between different programs which will reduce the accuracy of the measurement for the exact run time for each pass. This will add a small constant to both numerator and denominator of each ratio which will slightly dilute the results.
    b. The hardware specifications of the computer e.g. processor(s), memory, chip set, etc. will affect the run time of the program. However, the ratios of the run times (i.e. $\frac{t_{LS}}{t_{VS}}$, $\frac{t_{SS}}{t_{VS}}$ and $\frac{t_{LS}}{t_{VS}}$) should otherwise remain unaltered.
3. Program:
    a. The program is written in MATLAB and the run time is being measured through MATLAB commands for this purpose.
    b. A matrix of predetermined points is chosen on the surface of each fingerprint image. Each point that falls within the surface of the fingerprint will be used as a starting point and using each method the curve passing through that point (if any) will be extracted.
    c. The collective time for extracting all the curves for each fingerprint using each method is being measured.
    d. Using the measured data, $\frac{t_{LS}}{t_{VS}}$, $\frac{t_{SS}}{t_{VS}}$ and $\frac{t_{LS}}{t_{VS}}$ will be evaluated for each fingerprint. Eighty points for each ratio for each library will generate eighty data points for each ratio for each fingerprint library.
    e. The minimum, maximum, mean and standard deviation (.noted as σ) for each data set of each library will be evaluated.

The measurements are performed separately for each library in order to observe the variation degree in the results. If the results are in the same vicinity, similar outcome could be expected for any other set of fingerprints.

6.5 Simulation Results

Table 9 shows the results of the simulations carried out on each of the four libraries separately and also their combination.

Table 9: Run time comparison analogous to computation load comparison between LS, VS and SS methods

Lib #	$\frac{t_{LS}}{t_{VS}}$				$\frac{t_{SS}}{t_{VS}}$				$\frac{t_{SS}}{t_{LS}}$			
	Min	Max	Avg	σ	Min	Max	Avg	σ	Min	Max	Avg	σ
1	0.47	1.08	0.8	0.1	1.81	4.01	2.8	0.3	2.56	6.01	3.55	0.44
2	0.30	1.70	0.81	0.19	1.33	3.53	2.77	0.34	2.07	5.30	3.53	0.57
3	0.43	0.93	0.72	0.09	1.81	4.27	2.72	0.37	2.70	7.69	3.81	0.67
4	0.37	1.47	0.88	0.18	1.74	6.69	3.07	0.67	2.02	9.56	3.69	1.38
Com bined	0.30	1.70	0.80	0.16	1.33	6.69	2.84	0.47	2.02	9.56	3.65	0.85

By examining $\frac{t_{SS}}{t_{LS}}$ could be said that SS, which is the slowest method (smallest step size), compared to LS, which would be the fastest method (largest step size), on average takes 3.65 times longer to be executed. This is quite different than sixteen, which is the ratio of their step sizes (one and sixteen) for the reasons already described. Although, according to the simulation results, in certain cases it has taken 9.5 times longer to run SS versus LS. Deviation from average (σ) for this figure is quite large.

The average value of 0.8 for $\frac{t_{LS}}{t_{VS}}$ indicates that LS method on average is only 20% lighter to process than the VS method. Note once again that although LS has to be the fastest method because of using the longest step size, it is not suitable due to its lack of fidelity. The minimum value of $\frac{t_{LS}}{t_{VS}}$ is 0.3 which means a maximum of 70% time saving for LS vs. VS. Also, the maximum value is 1.7 which is an irregular case in which LS has taken 70% more than VS to run. The reason for that could be the straying away from the original curves and the following of lengthier curves, as anticipated for odd cases.

The standard deviation of 0.16 shows that in 68% of cases $\frac{t_{LS}}{t_{VS}}$ is between 0.64 and 0.96 which corresponds to 4% to 36% less computation using LS vs. VS. The result in each library is consistent with the overall outcome. Note that, when appropriate, the VS method will use the maximum step size and only will adjust to smaller step sizes when required. Using this strategy the extra computation load will be only added when it is necessary, for preserving the fidelity. The overhead of VS proved very small on average.

Eventually, the most important figure to compare the performance is $\frac{t_{SS}}{t_{VS}}$. SS is a viable method for extracting the curves but at the same time computationally it is quite expensive. The simulation result shows that on average the SS method takes 2.84 times the time of the VS method to run. Basically, on average the VS method reduces the computational load to almost one third of the SS method (more precisely, 1/2.84). The test results over separate libraries are consistent with the averages of $\frac{t_{SS}}{t_{VS}}$ on the combined set and are between 2.72 and 3.07. The time saving could be very significant when handling large numbers of fingerprints.

The overall standard deviation of 0.47 denotes that 68% of the results for $\frac{t_{SS}}{t_{VS}}$ are within 2.37 and 3.31. The saving achieved by using VS versus SS is quite significant and also is quite consistent when performed over separate libraries. Examining these methods on larger databases in the future could extend the proof of concept that was performed here.

CHAPTER 7 CONCLUSION AND FUTURE WORK

A comprehensive review of biometric security with a specific focus on fingerprint biometric is conducted and the relevant research progress and open issues in the field of fingerprint template protection are discussed. In this review, research progresses related to the fingerprint template protection have been presented. Furthermore, open issues in this area have been pointed out for further study.

It has been shown that both key binding i.e. fuzzy vault approach and irrevocable key generation from cancellable fingerprint template approach are desirable but they both suffer in addressing fingerprint features distortion at both registration and query stages. Cancellable approaches have the benefit of template revocability; however most of them fully depend on the accurate registration which is the most difficult challenge in any fingerprint identification system. Allegedly, fuzzy fingerprint vault could be a leading and state-of-the-art solution to protect fingerprint template. It addresses missing or fake minutiae effectively as it requires only enough genuine minutiae to unlock the secret key. It yet cannot solve the distortion problem and mostly it relies on accurate registration. Therefore, the approaches which are registration error tolerant or registration free are the best approaches in practical systems. It has been presented that a registration-free fuzzy vault approach could proficiently deal with the case of fake or missing minutiae and probably alignment problems. An entire fuzzy vault scheme was re-implemented as part of the work of this thesis.

As a separate and further contribution of this thesis, a novel approach for extracting groove lines or curves of a fingerprint was proposed. These curves which are one of the global features of fingerprints are used as *helper data* in fuzzy vault schemes and/or in classification of fingerprints. The simulation results show that with this approach the curvatures can be accurately extracted with optimum number of steps. As expected, the largest and more visible differences are in high curvature areas of the fingerprint. Evidently, the precision of the orientation field will affect the results regardless of fixing or changing the step size.

The proposed method which preserves the fidelity of the extracted curves is demonstrated to provide a noteworthy saving in computational load compared to the existing methods. The effect of this approach in improving overall fingerprint matching and/or classification will need to be the subjects of a later analysis.

As future work, the proposed method of variable step curve extraction could be incorporated in the fuzzy vault implementation. Integrating this method is expected to provide the required robustness to measurement noise and could additionally provide significant time saving which is a sought-after outcome for any pattern recognition and/or image processing application.

RESEARCH PUBLICATIONS

Mastali, N. 2012, ‘A Novel Approach for Curvature Detection in Global Fingerprint Feature Extraction’, *International Conference on Informatics, Electronics & Vision (ICIEV).*

Agbinya, J.I & Mastali, N. 2011, ‘ Design and Implementation of Multimodal Digital Identity Management System Using Fingerprint Matching and Face Recognition’, *6th International Conference on Broadband and Biomedical Communications.*

Mastali, N. 2011, ‘Fingerprint Template Protection: Development and Open Concerns’, *6th International Conference on Broadband and Biomedical Communications.*

Mastali, N. & Agbinya, J.I 2010, ‘Authentication of Subjects and Devices Using Biometrics and Identity Management Systems for Persuasive Mobile Computing: A Survey Paper’, *5th International Conference on Broadband and Biomedical Communications*.

REFERENCES

Acharya, T. & Ray, A.K. 2005, *Image processing: principles and applications* John Wiley & Sons, New Jersey.

Ahmad, F. & Mohamad, D. 2009, 'A Review on Fingerprint Classification Techniques', *Computer Technology and Development, 2009. ICCTD '09. International Conference on*, vol. 2, pp. 411-5.

Alonso-Fernandez, F., Fierrez-Aguilar, J. & Ortega-Garcia, J. 2005, 'An enhanced gabor filter-based segmentation algorithm for fingerprint recognition systems', *Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on*, IEEE, pp. 239-44.

AlTarawneh, M.S., Woo, W.L. & Dlay, S.S. 2008, 'Fuzzy Vault Crypto Biometric Key Based on Fingerprint Vector Features', *Communication Systems, Networks and Digital Signal Processing, 2008. CNSDSP 2008. 6th International Symposium on*, pp. 452-6.

Bala, D. 2008, 'Biometrics and information security', paper presented to the *Proceedings of the 5th annual conference on Information security curriculum development*, Kennesaw, Georgia.

Bhattacharyya, D., Ranjan, R., Das, P., Tai-hoon, K. & Bandyopadhyay, S.K. 2009, 'Biometric Authentication Techniques and its Future Possibilities', *Computer and Electrical Engineering, 2009. ICCEE '09. Second International Conference on*, vol. 2, pp. 652-5.

Boatwright, M. & Luo, X. 2007, 'What do we know about biometrics authentication?', paper presented to the *Proceedings of the 4th annual conference on Information security curriculum development*, Kennesaw, Georgia.

Boult, T.E., Schdrer, W.J. & Woodworth, R. 2007, 'Revocable Fingerprint Biotokens: Accuracy and Security Analysis', *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pp. 1-8.

Bovik, A.C. 2005, *Handbook of image and Video processing*, Elsevier, London, UK.

Cachin, C., Camenisch, J., Dodis, Y., Reyzin, L. & Smith, A. 2004, 'Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data', *Advances in Cryptology - EUROCRYPT 2004*, vol. 3027, Springer Berlin / Heidelberg, pp. 523-40.

Chenggang, Z. & Yingmei, S. 2009, 'Research about human face recognition technology', *Test and Measurement, 2009. ICTM '09. International Conference on*, vol. 1, pp. 420-2.

Chikkerur, S., Ratha, N.K., Connell, J.H. & Bolle, R.M. 2008, 'Generating Registration-free Cancelable Fingerprint Templates', *Biometrics: Theory, Applications and Systems, 2008. BTAS 2008. 2nd IEEE International Conference on*, pp. 1-6.

Cimato, S., Gamassi, M., Piuri, V., Sana, D., Sassi, R. & Scotti, F. 2006, 'Personal identification and verification using multimodal biometric data', *Computational Intelligence for Homeland Security and Personal Safety, Proceedings of the 2006 IEEE International Conference on*, pp. 41-5.

Clancy, T.C., Kiyavash, N. & Lin, D.J. 2003, 'Secure smartcardbased fingerprint authentication', paper presented to the *Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications*, Berkley, California.

Cui, W., Wu, G., Hua, R. & Yang, H. 2008, 'The research of edge detection algorithm for Fingerprint images', *Automation Congress, 2008. WAC 2008. World*, IEEE, pp. 1-5.

Dacheng, X. & Xiaotao, W. 2010, 'A scheme for cancelable fingerprint fuzzy vault based on chaotic sequence', *Mechatronics and Automation (ICMA), 2010 International Conference on*, pp. 329-32.

Daesung, M., Sungju, L., Yongwha, C., Sung Bum, P. & Kiyoung, M. 2008, 'Implementation of automatic fuzzy fingerprint vault', *Machine Learning and Cybernetics, 2008 International Conference on*, vol. 7, pp. 3781-6.

Dass, S.C. 2004, 'Markov random field models for directional field and singularity extraction in fingerprint images', *Image Processing, IEEE Transactions on*, vol. 13, no. 10, pp. 1358-67.

Dass, S.C. & Jain, A.K. 2004a, 'Fingerprint classification using orientation field flow curves', paper presented to the *Computer Vision, Graphics and Image Processing* Kolkata, India, December.

Dass, S.C. & Jain, A.K. 2004b, 'Markov random field models for directional field and singularity extraction in fingerprint images', *Computer Vision, Graphics and Image Processing* Kolkata, India.

Davida, G.I., Frankel, Y. & Matt, B.J. 1998, 'On enabling secure applications through off-line biometric identification', *Security and Privacy, 1998. Proceedings. 1998 IEEE Symposium on*, pp. 148-57.

De Luis-García, R., Alberola-López, C., Aghzout, O. & Ruiz-Alzola, J. 2003, 'Biometric identification systems', *Signal Processing*, vol. 83, no. 12, pp. 2539-57.

de Luis-García, R., Alberola-López, C., Aghzout, O. & Ruiz-Alzola, J. 2003, 'Biometric identification systems', *Signal Processing*, vol. 83, no. 12, pp. 2539-57.

Dolezel, M., Hejtmankova, D., Busch, C. & Drahansky, M. 2010, 'Fingerprint Area Detection in Fingerprint Images Based on Enhanced Gabor Filtering', *Database Theory and Application, Bio-Science and Bio-Technology*, pp. 234-40.

Farooq, F., Bolle, R.M., Tsai-Yang, J. & Ratha, N. 2007, 'Anonymous and Revocable Fingerprint Recognition', *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pp. 1-7.

Feng, D., Lin, D., Yung, M., Chung, Y., Moon, D., Lee, S., Jung, S., Kim, T. & Ahn, D. 2005, 'Automatic Alignment of Fingerprint Features for Fuzzy Fingerprint Vault', *Information Security and Cryptology*, vol. 3822, Springer Berlin / Heidelberg, pp. 358-69.

Fengling, H., Jiankun, H., Leilei, H. & Yi, W. 2007, 'Generation of Reliable PINs from Fingerprints', *Communications, 2007. ICC '07. IEEE International Conference on*, pp. 1191-6.

Guo Xiao, Q. & Hu Ai, Q. 2009, 'The Automatic Fuzzy Fingerprint Vault Based on Geometric Hashing: Vulnerability Analysis and Security Enhancement', *Multimedia Information Networking and Security, 2009. MINES '09. International Conference on*, vol. 1, pp. 62-7.

Han, F., Yu, X. & Hu, J. 2005, 'A new way of generating grid-scroll chaos and its application to biometric authentication', *Industrial Electronics Society, 2005. IECON 2005. 31st Annual Conference of IEEE*, p. 6 pp.

Harne, S., Satao, K.J. & Khan, M.I. 2011, 'Biometric authentication using minutiae fingerprint.(Report)', *International Journal of Research and Reviews in Computer Science*, vol. 2, no. 3, p. 761(8).

Hu, J. 2008, 'Mobile fingerprint template protection: Progress and open issues', *Industrial Electronics and Applications, 2008. ICIEA 2008. 3rd IEEE Conference on*, pp. 2133-8.

Isalam, R. 2007, 'Digital identity modelling and analysis'', University of Technology, Sydney

Jain, A., Lin, H. & Bolle, R. 1997, 'On-line fingerprint verification', *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, no. 4, pp. 302-14.

Jain, A.k., Bolle, R. & Pankanti, S. (eds) 1999, *Biometrics: personal identification in networked security*, Springer, Massachusetts.

Jones, L.A., Anton, A.I. & Earp, J.B. 2007, 'Towards understanding user perceptions of authentication technologies', paper presented to the *Proceedings of the 2007 ACM workshop on Privacy in electronic society*, Alexandria, Virginia, USA.

Juels, A. & Sudan, M. 2002, 'A fuzzy vault scheme', *Information Theory, 2002. Proceedings. 2002 IEEE International Symposium on*, p. 408.

Juels, A. & Wattenberg, M. 1999, 'A fuzzy commitment scheme', paper presented to the *Proceedings of the 6th ACM conference on Computer and communications security*, Kent Ridge Digital Labs, Singapore.

Kai, X. & Jiankun, H. 2009, 'Biometric Mobile Template Protection: A Composite Feature Based Fingerprint Fuzzy Vault', *Communications, 2009. ICC '09. IEEE International Conference on*, pp. 1-5.

Kanade, T., Jain, A., Ratha, N., Uludag, U. & Pankanti, S. 2005, 'Fuzzy Vault for Fingerprints', *Audio- and Video-Based Biometric Person Authentication*, vol. 3546, Springer Berlin / Heidelberg, pp. 55-71.

Kotlarchyk, A., Pandya, A. & Zhuang, H. 2008, 'Simulation and experimental studies on fuzzy vault fingerprint cryptography', *International Journal of Knowledge-based and Intelligent Engineering Systems*, vol. 12, no. 5, pp. 305-17.

Lalithamani, N. & Soman, K.P. 2009a, 'An Efficient Approach for Non-Invertible Cryptographic Key Generation from Cancelable Fingerprint

Biometrics', *Advances in Recent Technologies in Communication and Computing, 2009. ARTCom '09. International Conference on*, pp. 47-52.

Lalithamani, N. & Soman, K.P. 2009b, 'Towards generating irrevocable key for cryptography from cancelable fingerprints', *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*, pp. 563-8.

Lin, H., Yifei, W. & Jain, A. 1998, 'Fingerprint image enhancement: algorithm and performance evaluation', *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 8, pp. 777-89.

Malickas, A. & Vitkus, R. 1999, 'Fingerprint Registration Using Composite Features Consensus', *INFORMATICA*, vol. 10, no. 4, pp. 389-402.

Maltoni, D., Maio, D., Jain, A.k. & Prabhakar, S. 2003, *Handbook of Fingerprint Recognition*, Springer, New York

Menezes, A.J., Oorschot, P.V. & Vanstone, S.A. 1996, *Handbook of applied cryptography*, CRC Press LLC, Unites States of America, NY.

Mil'shtein, S., Pillai, A., Shendye, A., Liessner, C. & Baier, M. 2008, 'Fingerprint Recognition Algorithms for Partial and Full Fingerprints', *Technologies for Homeland Security, 2008 IEEE Conference on*, pp. 449-52.

Nandakumar, K., Jain, A.K. & Pankanti, S. 2007, 'Fingerprint-Based Fuzzy Vault: Implementation and Performance', *Information Forensics and Security, IEEE Transactions on*, vol. 2, no. 4, pp. 744-57.

Nikam, S.B. & Agarwal, S. 2008, 'Level 2 features and wavelet analysis based hybrid fingerprint matcher', paper presented to the *Proceedings of the 1st Bangalore Annual Compute Conference*, Bangalore, India.

O'Gorman, L. & Nickerson, J.V. 1989, 'An approach to fingerprint filter design', *Pattern Recognition*, vol. 22, no. 1, pp. 29-38.

Rao, A.R. 1990, *A taxonomy for texture description and identification*, Springer-Verlag, New York.

Ratha, N.K., Chen, S. & Jain, A.K. 1995, 'Adaptive flow orientation-based feature extraction in fingerprint images', *Pattern Recognition*, vol. 28, no. 11, pp. 1657-72.

Ratha, N.K., Chikkerur, S., Connell, J.H. & Bolle, R.M. 2007, 'Generating Cancelable Fingerprint Templates', *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 4, pp. 561-72.

Sabena, F., Dehghantanha, A. & Seddon, A.P. 2010, 'A Review of Vulnerabilities in Identity Management Using Biometrics', *Future Networks, 2010. ICFN '10. Second International Conference on*, pp. 42-9.

Sashank Singhvi, R., Venkatachalam, S.P., Kannan, P.M. & Palanisamy, V. 2009, 'Cryptography key generation using biometrics', *Control, Automation, Communication and Energy Conservation, 2009. INCACEC 2009. 2009 International Conference on*, pp. 1-6.

Scheirer, W.J. & Boult, T.E. 2007, 'Cracking Fuzzy Vaults and Biometric Encryption', *Biometrics Symposium, 2007*, pp. 1-6.

Sen, W., Wei , Z. & Yang Sheng, W. 2002, 'Fingerprint classification by directional fields', *Multimodal Interfaces, 2002. Proceedings. Fourth IEEE International Conference on*, pp. 395-9.

Shen, L.L., Kot, A. & Koo, W.M. 2001, 'Quality measures of fingerprint images', *Audio-and Video-based Biometric Person Authentication*, Springer, pp. 266-71.

Shenglin, Y. & Verbauwhede, I. 2005, 'Automatic secure fingerprint verification system based on fuzzy vault scheme', *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, vol. 5, pp. v/609-v/12 Vol. 5.

Shukla, A. & Tiwari, R. 2008, 'Intelligent Biometric System: A Case Study', *Information Technology Research*, vol. 1, no. 3, pp. 41-56

Singla, R. & Saini, K. 2009, 'Application of Fingerprint Recognition in Process Control', *Image and Signal Processing, 2009. CISP '09. 2nd International Congress on*, pp. 1-5.

Stallings, W. (ed.) *2003, Cryptography and Network Security: Principles and Practices, 3 edn, Prentice Hall, New Jersey.*

Sungju, L., Daesung, M., Hanna, C. & Yongwha, C. 2008, 'Memory-Efficient Fuzzy Fingerprint Vault based on the Geometric Hashing', *Information Security and Assurance, 2008. ISA 2008. International Conference on*, pp. 312-5.

Sungju, L., Daesung, M., Woo Yong, C. & Yongwha, C. 2008, 'Analysis of Tradeoffs among Verification Accuracy, Memory Consumption, and Execution Time in the GH-Based Fuzzy Fingerprint Vault', *Security Technology, 2008. SECTECH '08. International Conference on*, pp. 75-8.

Tistarelli, M., Bigun, J. & Grosso, E. (eds) 2005, Advanced Studies in Biometrics, vol. 3116, Springer, Berlin.

Uludag, U. & Anil, J. 2006, 'Securing Fingerprint Template: Fuzzy Vault with Helper Data', *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW '06. Conference on*, pp. 163-.

Uludag, U. & Jain, A. 2006, 'Securing Fingerprint Template: Fuzzy Vault with Helper Data', paper presented to the *Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*.

Wang, L., Bhattacharjee, N., Gupta, G. & Srinivasan, B. 2010, 'Adaptive approach to fingerprint image enhancement', *Proceedings of the 8th International Conference on Advances in Mobile Computing and Multimedia*, ACM, pp. 42-9.

Wayman, J.L. 2008, 'Biometrics in Identity Management Systems', *Security & Privacy, IEEE*, vol. 6, no. 2, pp. 30-7.

Wu, Y., He, G., Zhang, X. & Liu, Z. 2007, 'A Fast Fingerprint Identification Pre-Processing Algorithm', *Bioinformatics and Biomedical Engineering, 2007. ICBBE 2007. The 1st International Conference on*, pp. 596-8.

Yang, G., Zhou, G.T., Yin, Y. & Yang, X. 2010, 'K-means based fingerprint segmentation with sensor interoperability', *EURASIP Journal on Advances in Signal Processing*, vol. 2010, p. 54.

Yang, S. & Verbauwhede, I.M. 2003, 'A secure fingerprint matching technique', paper presented to the *Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications*, Berkley, California.

Yuan, L. 2009, 'Application of Fuzzy Pattern Recognition on Fingerprint Processing', *Intelligent Computation Technology and Automation,* *2009. ICICTA'09. Second International Conference on*, vol. 2, IEEE, pp. 660-3.

Zhang, Y., Brady, M. & Smith, S. 2001, 'Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm', *IEEE Trans Med Imaging*, vol. 20, no. 1, pp. 45-57.