A

MAJOR REPORT

ON

INCREASING SECURITY AND OPTIMIZATION USING FINGERPRINT AND CLOUD COMPUTING IN ONLINE VOTING SYSTEM

Submitted in Partial fulfillment of the requirement
For the award of the degree of

MASTER OF TECHNOLOGY
(INFORMATION SYSTEM)

Submitted by
PRAVESH KUMAR BANSAL
(07/IS/09)
Under the Guidance of
RITU AGARWAL
(Asst. Professor)
Dept. of Information Technology

DELHI TECHNOLOGICAL UNIVERSITY
(DEPARTMENT OF INFORMATION TECHNOLOGY)
BAWANA ROAD, DELHI-110042

SESSION: 2009-11

CERTIFICATE

This is to certify that the MAJOR REPORT titled as "**INCREASING SECURITY AND OPTIMIZATION USING FINGERPRINT AND CLOUD COMPUTING IN ONLINE VOTING SYSTEM**" is being submitted by **Pravesh Kumar Bansal** Class roll no. 07/IS/09, in partial fulfillment for the award of "Master of Technology Degree in Information System" in Delhi Technological University, Delhi; is the original work carried out by him under the guidance and supervision. The matter contained in this report has not been submitted elsewhere for reward for any other degree.

(Project Guide)
Ritu Agarwal
Assistant Professor
Dept. Of Information Technology
Delhi Technological University
Bawana Road, Delhi-110042

ACKNOWLEDGEMENT

Firstly, I would like to express my hearty gratitude and thanks to my project guide **RITU AGARWAL**, Asst. Prof., Information Technology Dept., D.T.U. Delhi. For continuous inspiration, encouragement and guidance in every stage of preparation of this project report.

I am extremely thankful to Prof. **O.P.VERMA**, Head of Dept., Information Technology, D.T.U., Delhi, for the support provided by him during the entire duration of degree course and especially in this thesis. I am thankful to all teaching and non-teaching staff at D.T.U., and my fellows, who have helped me directly or indirectly in completion of this thesis report.

PRAVESH KUMAR BANSAL
M.TECH (INFORAMTION SYSTEM)
ROLL NO. 07/IS/09
EMAILID: bansal086@gmail.com

ABSTRACT

Election is a process in which voters choose their representatives and express their preferences for the way that they will be governed. Using the decade old voting system to collect votes is no longer considered efficient due to the various recurring errors. The advancement of information and telecommunications technologies allow for a fully automated online computerized election process. An electronic voting system defines rules for valid voting and gives an efficient method of counting votes, which are aggregated to yield a final result. Correctness, robustness to fraudulent behaviors, coherence, consistency, security, and transparency of voting are all key requirements for the integrity of an election process. Moreover, electronic voting systems can improve voter identification process by utilizing biometric recognition which provides more security. Biometrics is becoming an essential component of personal identification solutions, since biometric identifiers cannot be shared or misplaced, and they represent any individual's identity. Fingerprint matching is a significant part of this process.

This project work propose online security enhancement technique using level 3 fingerprint features and scale invariant feature transform algorithm for matching purpose.

Firstly fingerprint of good quality are acquired by using optical scanner. Image normalization is done using Gaussian blurring and sliding window contrast adjustment. Pores are extracted and estimated. Using these estimated pores, matching is done from template database to stored database using SIFT algorithm. Scale Invariant Features Transform (SIFT) is an algorithm in computer vision to detect and describe local features in images. The features are invariant to image scaling and rotation. They are well localized in both the spatial and frequency domains. The features are highly distinctive, which allows a single feature to be correctly matched with high probability against a large database of features, providing a basis for object and scene recognition.

The voter's fingerprint database is huge in size. To check the authentication of voter, input fingerprint template has to be matched against entire stored fingerprint database. To speedup the process and gain efficiency, fingerprint database will be uploaded on cloud where matching results can be retrieved fast as compare to matching on a single high computing power CPU.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

In an increasingly digital world, reliable personal authentication has become an important human computer interface activity. National security, e-commerce, and access to computer networks are some examples where establishing a person's identity is vital. Existing security measures rely on knowledge-based approaches like passwords or token-based approaches such as swipe cards and passports to control access to physical and virtual spaces.

Though ubiquitous, such methods are not very secure. Tokens such as badges and access cards may be shared or stolen. Passwords and Personal Identification Number (PIN) numbers may be stolen electronically. Furthermore, they cannot differentiate between authorized user and a person having access to the tokens or knowledge. Biometrics such as fingerprint, face and voice print offers means of reliable personal authentication that can address these problems and is gaining citizen and government acceptance.

1.1 Problem Definition

Fingerprint recognition is a complex pattern recognition problem. It is difficult to design accurate algorithms capable of extracting salient features and matching them in a robust way, especially in poor quality fingerprint images and when low-cost acquisition devices with small area are adopted. There is a popular misconception that automatic fingerprint recognition is a fully solved problem since it was one of the first applications of machine pattern recognition. On the contrary, fingerprint recognition is still a challenging and important pattern recognition problem.

The real challenge is matching fingerprints affected by:

i) High displacement/or rotation which results in smaller overlap between template and query fingerprints (this case can be treated as similar to matching partial fingerprints).

ii) Non-linear distortion caused by the finger plasticity.

iii) Different pressure and skin condition

iv) Feature extraction errors which may result in spurious or missing features. The vast majority of contemporary automated fingerprint authentication systems (AFAS) are minutiae (level 2 features) based [1]. Minutiae-based systems generally rely on finding

correspondences between the minutia points present in “query” and “reference” fingerprint images.

These systems normally perform well with high quality fingerprint images and a sufficient fingerprint surface area. These conditions, however, may not always be attainable. In many cases, only a small portion of the “query” fingerprint can be compared with the “reference” fingerprints. As a result, the number of minutiae (Level 2 feature of fingerprint) correspondences might be significantly decreases and the matching algorithm would not be able to make a decision with high certainty. The description of minutiae is given in background (Chapter 2). This effect is even more marked on intrinsically poor quality fingers, where only a subset of the minutiae can be extracted and used with sufficient reliability. Although minutiae may carry most of the fingerprint’s discriminatory information, they do not always constitute the best trade-off between accuracy and robustness.

This has led the designers of fingerprint recognition techniques to search for other fingerprint distinguishing features, beyond minutiae, which may be used in conjunction with minutiae (and not as an alternative) to increase the system accuracy and robustness. It is a known fact that the presence of Level 3 features in fingerprints provides minutiae detail for matching and the potential for increased accuracy. The forensic experts in law enforcement often make use of Level 3 features, such as sweat pores and ridge contours, to compare fingerprint samples when insufficient minutia points are present in the fingerprint image or poor image quality hampers minutiae analysis. That is, experts take advantage of an extended feature set in order to conduct a more effective matching.

Despite their discriminating property, level 3 features are barely utilized in the commercial automated fingerprint authentication systems (AFAS), as a result a large amount of fingerprint information is ignored by such systems. This is mainly because, most of these authentication systems are equipped with 500ppi (pixels per inch) scanners, and reliably (or consistently) extracting, “fine and detailed” Level 3 features require high resolution images. While this may have been the case with many of the older live-scan devices, the current devices are capable of detecting a reasonable amount of level three details even at the relatively limited 500ppi resolution.

Ray et al. [2] have presented a means of modeling and extracting pores (which are considered as highly distinctive Level 3 features) from 500ppi fingerprint images. This study showed that while

not every fingerprint image obtained with a 500ppi scanner has evident pores, a substantial number of them do have. Thus, it is a natural step to try to extract Level 3 information, and use them to achieve robust matching decisions. In addition, the fine details of level 3 features could potentially be exploited in circumstances that require high-confidence matches.

1.2 Objective

The objective of this thesis is to improve the fingerprint matching properties using SIFT pores matching technique.

In addition, this thesis moves forward in order to fulfill the following objectives:

1. To design fingerprint Level 3 feature extraction method and match using SIFT algorithm.
2. To analyze the results by experimenting on multiple images.

1.3 Methodology

The work addresses various issues of challenges (discussed in previous section) in fingerprint matching. The aim is to reduce the error rates, namely False Acceptance Rate (FAR) and False Rejection Error (FRR) in the existing fingerprint matching algorithms. The proposed approach utilizes Level 3 features (pores) for matching fingerprints at 500 pixels per inch (ppi).

The proposed approach addresses the various challenges in fingerprint matching in following way:

1. The plastic nature of finger skin results in non-linear distortion in consecutive acquisitions of the same finger. To deal with this problem the matching of all features (Level 3) are done within a local region. The use of localized matching minimizes the effects of non-linear distortion. This is because the effects of distortion do not radically alter the fingerprint pattern locally.

2. Due to high displacement and rotation which are introduced during fingerprint acquisition, different impressions of the same finger differ from each other. Most of the existing minutia matching algorithms first align fingerprint images and then find minutia correspondences. But in case of poor quality fingerprints, global registration (alignment) parameters do not exist and as a result it is not possible to get a correct alignment. The errors introduced during registration steps can introduce errors in the subsequent steps. The proposed approach does not use alignment at any stage and relies on rotationally invariant structures and features.

3. The high displacement/rotation introduced during fingerprint acquisition results in “small

overlap" between query and reference fingerprints. Also the noise introduced by several factors such as pore or skin conditions, unclean scanner surface etcetera, results in a very small portion of the fingerprint which can actually be used for comparison. The use of Level 3 features takes care of such situations. Studies show that given a sufficiently high resolution fingerprint, the use of Level 3 features from fingerprint fragments results in same quantity of discriminative information that can be extracted when Level 2 features are considered and the entire image is used.

4. Finally, due to the noise in the fingerprints the feature extraction techniques often introduce errors such as missing or spurious minutiae/pores. The matching technique should handle such cases. The proposed approach uses an SIFT matching algorithm for pores matching, which can accommodate perturbations of minutiae/pores from their true locations and can tolerate spurious and missing minutiae/pores.

1.4 Thesis outline

This thesis consists of eight chapters with references. The outline about each chapter is given below:

Chapter 1 introduction to biometric basics and description of research objectives.

Chapter 2 describes the background of fingerprint biometric system.

Chapter 3 literature survey and work done by several authors and researchers.

Chapter 4 describes the fingerprint level 3 features.

Chapter 5 describes the scale invariant feature transform algorithm.

Chapter 6 describes the work done.

Chapter 7 analyzing the results.

Chapter 8 concludes the whole report and future scope is also suggested.

CHAPTER 2

BACKGROUND

This chapter describes general concept of fingerprint verification system. A biometric authentication system operates by acquiring biometric data from a user and comparing it against the template data stored in a database in order to identify a person or to verify a claimed identity. Most systems store multiple templates per user in order to account for variations observed in a person's biometric data.

A biometric authentication system uses the physiological (fingerprints, face, hand geometry, iris) and/or behavioral (voice, signature, keystroke dynamics) traits of an individual to identify a person or to verify a claimed identity [11]. A typical biometric system operates in two distinct stages: the enrollment stage and the authentication stage. During enrollment, a user's biometric data (e.g., fingerprints) is acquired and processed to extract a feature set (e.g., minutiae points) that is stored in the database. The stored feature set, labeled with the user's identity, is referred to as a template. In order to account for variations in the biometric data of a user, multiple templates corresponding to each user may be stored. During authentication, a user's biometric data is once again acquired and processed, and the extracted feature set is matched against the template(s) stored in the database in order to identify a previously enrolled individual or to validate a claimed identity.

The matching accuracy of a biometrics-based authentication system relies on the stability (durability) of the biometric data associated with an individual over time. In reality, however, the biometric data acquired from an individual is susceptible to changes due to improper interaction with the sensor (e.g., partial fingerprints, change in pose during face-image acquisition), modifications in sensor characteristics (e.g., optical vs. solid-state fingerprint sensor), variations in environmental factors (e.g., dry weather resulting in faint fingerprints) and temporary alterations in the biometric trait itself (e.g., cuts/scars on fingerprints). In other words, the biometric measurements tend to have large intra-class variability. Thus, it is possible for the stored template data to be significantly different and the storage and computational overheads introduced by multiple templates. For an efficient functioning of a biometric system, the process

of template selection has to be automated. However, there is limited literature dealing with the problem of automatic template selection in a biometric system.

2.1 Problem with cards and PIN based authentication

PIN's (Personal Identification Numbers) were one of the first identifiers to offer automated recognition. However, it should be understood that this means recognition of the PIN, not necessarily recognition of the person who has provided it. The same applies with cards and other tokens. It may easily recognize the token, but token can be accessible by anybody. Using the two together provides a slightly higher confidence level, but this is still easily compromised if one is determined to do so.

A biometrics however cannot be easily transferred between individuals and represents as unique an identifier as which is likely to be seeing. If it can be able to automate the verification procedure in a user-friendly manner, there will extensive scope for integrating biometrics into a variety of processes.

The keys are usually stored in a secure location (e.g., tamper-resistant hardware) and password-based authentication is commonly used for controlling access to cryptographic keys. However, passwords can be easily lost, stolen, forgotten or guessed using social engineering and dictionary attacks. Limitations of password-based authentication can be alleviated by using stronger authentication schemes such as biometrics. Biometric systems establish the identity of a person based on his/her anatomical or behavioral traits such as face, fingerprint, iris, voice, etc. Biometric authentication is more reliable than password-based authentication because biometric traits cannot be lost or forgotten and it is difficult to share or forge these traits.

2.2 The move from forensics to civil application.

The early applications of Biometric technology were limited to the area of forensics. These original applications of fingerprint, which relied on images from inked ten-print cards that were captured by digital cameras, increased not only the speed of the identification response, but also the level of accuracy. The criminal and civil systems differ in terms of their complexity and cost because of their differing purposes. Additionally, while the search databases for forensic applications are maintained for law enforcement purposes, the databases for civil applications are operated and maintained by non-law enforcement personnel. The requirements for record

retention, confidentiality, and even accuracy can be very different for civil applications. The use of fingerprint technology in civil applications, particularly public benefits programs, has not been without criticism.

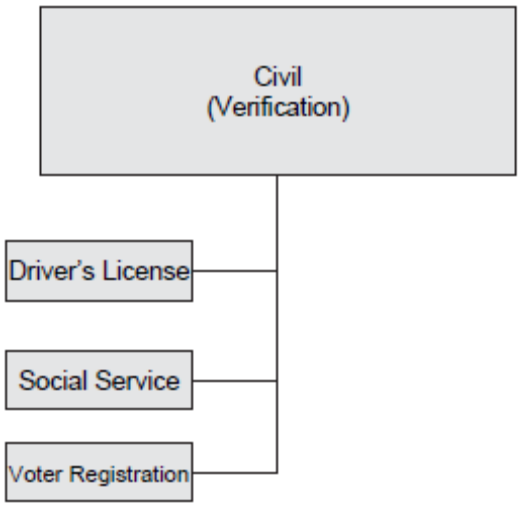


Figure 2.1 Civil usage of fingerprint technology

2.3 General fingerprint verification system.

A fingerprint-based biometric system can be logically divided into two distinct operational phases: enrollment, and recognition as shown in Figure 2.2. A general flow model for fingerprint matching describes its processing steps.

During the enrollment phase the fingerprint features of an individual are collected in order to create a compact representation, called reference template. The reference template is usually stored in a system database, along with other identification information of the enrolled person. The enrollment phase can take place in a controlled environment where the person to be enrolled may be assisted by an expert. The time consumption is less important than the need to create an accurate and complete reference template. In practice, the acquired reference templates must pass a quality check in order to be stored in the system database.

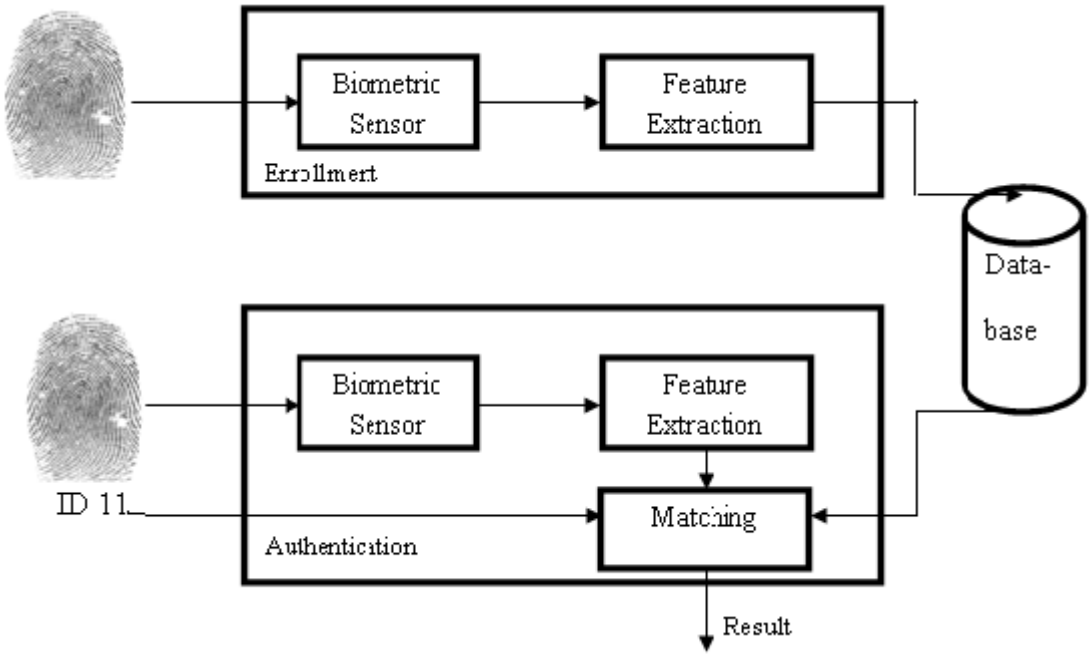


Figure 2.2 Common phases of fingerprint biometric system

During the recognition phase the features extracted from a single fingerprint impression are used to produce a similar representation as the reference template. The new representation, called test template, is fed into a fingerprint matching algorithm that compares it against one, or more reference templates stored in the system database in order to verify, or retrieve the user's identity. General architecture of a fingerprint verification system smoothening. However, it is to be noted that unlike regular images, the fingerprint image represents a system of oriented texture and has very rich structural information within the image. Furthermore, the definition of noise and unwanted artifacts are also specific to fingerprints. The fingerprint image enhancement algorithms are specifically designed to exploit the periodic and directional nature of the ridges. Finally, the minutiae features are extracted from the image and are subsequently used for matching. Although research in fingerprint verification has been pursued for several decades now, there are several open research challenges still remaining, some of which will be addressed in the ensuing sections of this thesis.

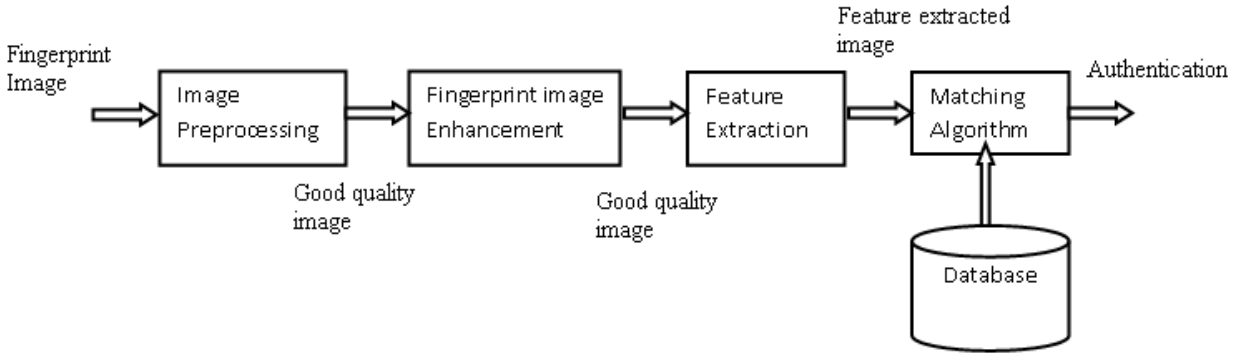


Figure 2.3 General architecture of Fingerprint verification system.

The various stages of a typical fingerprint recognition system is shown in Figure 2.3. Fingerprint verification is to verify the authenticity of one person by his/her fingerprint. The user provides his/her fingerprint together with his/her identity information like his/her ID number. The fingerprint verification system retrieves the fingerprint template according to the ID number and matches the template with the real-time acquired fingerprint from the user. Usually it is the underlying design principle of AFAS (Automatic Fingerprint Authentication System).

Fingerprint identification is to specify one person's identity by his fingerprint(s). Without knowledge of the person's identity, the fingerprint identification system tries to match his fingerprint(s) with those in the whole fingerprint database. It is especially useful for criminal investigation cases. And it is the design principle of AFIS (Automatic Fingerprint Identification System).

However, all fingerprint recognition problems, either verification or identification, are ultimately based on a well-defined representation of a fingerprint. As long as the representation of fingerprints remains the uniqueness and keeps simple, the fingerprint matching, either for the 1-to-1 verification case or 1-to-m identification case, is straightforward and easy.

2.4 Fingerprint pattern classification.

The fingerprint surface is made up of a system of ridges and valleys that serve as friction surface when gripping the objects. The surface exhibits very rich structural information when examined as an image. The fingerprint images can be represented by both global as well as local features. The global features include the ridge orientation, ridge spacing and singular points such as core and delta. However, verification usually relies entirely on minutiae features. Minutiae are local

features marked by ridge discontinuities. There are about 18 distinct types of minutiae features that include ridge endings, bifurcations, crossovers and islands. Among these, ridge endings and bifurcation are the commonly used features. Some of fingerprint classes are shown in Figure 2.4. A ridge ending occurs when the ridge flow abruptly terminates and a ridge bifurcation is marked by a fork in the ridge flow. Most matching algorithms do not even differentiate between these two types since they can easily get exchanged under different pressures during acquisition. Global features do not have sufficient discriminative power on their own and are therefore used for binning or classification before the extraction of the local minutiae features.

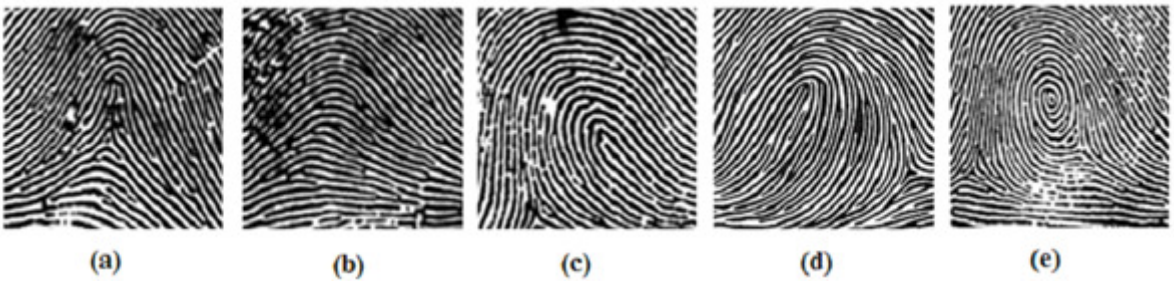


Figure 2.4 Fingerprint Classes (a) Tended Arch (b) Arch (c) Right Loop (d) Left Loop (e) Whorl.

2.4.1 Pattern Classification:

The classification of fingerprints into distinct groups based on general similarities allows the fingerprint examiner to search for an unidentified fingerprint within a specific section of the fingerprint file rather than having to search the whole file. There are numerous fingerprint classification systems in use throughout the world today. These systems are all based on three fundamental ridge formations described by Purkinje, Galton, Vucetich and Henry. They are the arch, the loop - radial and ulnar, and the whorl.

Branch and end points of epidermal ridges were used by Sir Francis Galton in 1872 to develop a probabilistic model of fingerprint individuality, and they have been used since then in both forensic (Cummins and Midlo, 1943) and automated matching (Blue, Candela, Gruther, Chellapa, and Wilson, 1994; Hrechak and McHugh, 1990). These Galton features, or minutiae, contain unique information that enables their use in probabilistic analyses. Each Galton feature has a specific type, i.e., branch point or end point, a unique location on the fingerprint, and a specific orientation (Stoney and Thornton, 1986). The orientation can be defined for an end point, for example, as the approximate tangent angle to the ridge ending. Most probabilistic

models to date have utilized Galton features exclusively. The first model, published in 1977 by James Osterburg, et al at the University of Illinois, determines the probability of occurrence of a certain configuration of Galton features in a fingerprint. Two years later, a member of Osterburg's team, Stanley Sclove, published a paper presenting the occurrence of Galton features as a two-dimensional Markov model. Both of these models can be adapted to use pores instead of Galton features. Pores have been used historically to assist in forensic matching. Although most matching methods have emphasized minutia comparisons and used pores as ancillary comparison features, the ability to match prints based on pore information alone has been documented (Ashbaugh, 1983; Locard, 1912; Stosz and Alyea, 1994). The concept of using pores to match prints has been essentially dormant during the rise of automated fingerprint recognition systems [18].

2.4.2 Fingerprint Represtation:

The purpose of the matching algorithm is to compare two fingerprint images or templates and returns a similarity score that corresponds to the probability of match between the two prints. Except for the correlation based algorithms, most of the algorithm extract features for the purpose of matching. Minutiae features are the most popular of all the existing representation and also form the basis of the visual matching process used by human experts. Minutiae represent local discontinuities and mark position where the ridge comes to an end or bifurcates into two (See figure 2.5). These form the most frequent types of minutiae , although a total of 18 minutiae types have been identified so far [17]. Each minutiae may be described by a number of attributes such as its position (x,y) its orientation θ, its quality.

2.5 Template selection techniques:

The goal of the new generation of fingerprint technique is to support the matching with Level 3 features, increasing the system security to the governmental and Police levels. Here there are three template selection criteria, being minutiae-based, and correlation-based, and coherence based.

2.5.1 Minutiae Acquisition Technique

Most of the finger-scan technologies are based on minutiae. Minutia-based techniques represent the fingerprint by its local features, like terminations and bifurcations. This approach has been intensively studied, and is the backbone of the current available fingerprint recognition products.

This work also concentrates on the same approach. However a drawback of this technique is that it suffers from most of the problems of minutiae-based systems. Still, many false minutiae are extracted, causing at least a part of the templates to be rather unreliable.

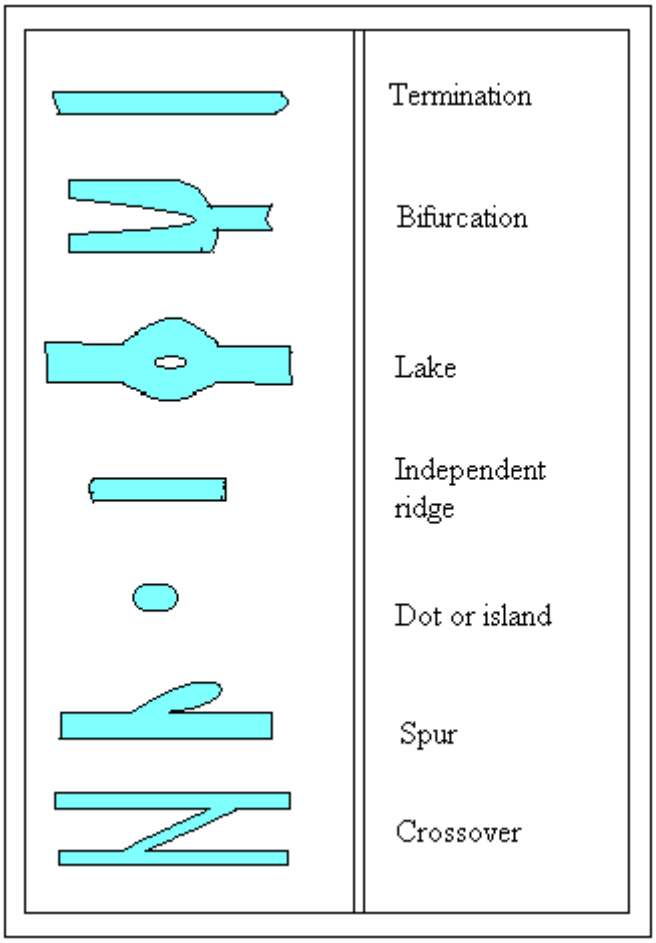


Figure 2.5 Different minutiae

2.5.2 Correlation-Based Template Selection:

The second method satisfies the template requirements most directly. In this method, templates are selected by checking how well they fit at other locations in the same fingerprint. If a template fits almost as well at another location as it does at its original location, it is not a useful template. However, if a template fits much worse at all other locations in the fingerprint, it is a template that offers a lot of distinction. Therefore, the ratio of fit at a template's original location to the fit at the next best location can be used as a template selection criterion.

Since the correlation-based checking is carried out by means of template matching, this method consumes a lot of computational power. This makes it a less attractive method to use. However,

it is for instance possible to combine this approach with the other two methods. In that case, possible template locations are extracted by one of the methods of the previous subsections. Then, the correlation characteristics of those locations are checked as an additional selection criterion.

2.5.3 Coherence-Based Template Selection

The coherence of an image area is a measure that indicates how well the local gradients are pointing in the same direction. In areas where the ridge-valley structures are only parallel lines, the coherence is very high, while in noisy areas, the coherence is low. Templates that are chosen in regions of high coherence values cannot be located reliably in a second fingerprint. However, at locations around minutiae, more grayscale gradient orientations are present, resulting in a significantly lower coherence. Therefore, the coherence can be used as an appropriate measure that indicates the presence of minutiae as well as a measure that indicates how well a template can be located in the secondary fingerprint.

2.6 General approach for fingerprint recognition

Fingerprints are imprints formed by friction ridges of the skin and thumbs. They have long been used for identification because of their immutability and individuality. Immutability refers to the permanent and unchanging character of the pattern on each finger. Individuality refers to the uniqueness of ridge details across individuals; the probability that two fingerprints are alike is about 1 in 1.9x1015. However, manual fingerprint verification is so tedious, time consuming and expensive that is incapable of meeting today's increasing performance requirements. An automatic fingerprint identification system is widely adopted in many applications such as building or area security and ATM machines. The general approach will be described for fingerprint recognition:

2.6.1 Approach: Based on minutiae located in a fingerprint

Most automatic systems for fingerprint comparison are based on minutiae matching. Minutiae are local discontinuities in the fingerprint pattern. In practice only ridge ending and ridge bifurcation minutiae types are used in fingerprint recognition as shown in Figure 2.6.

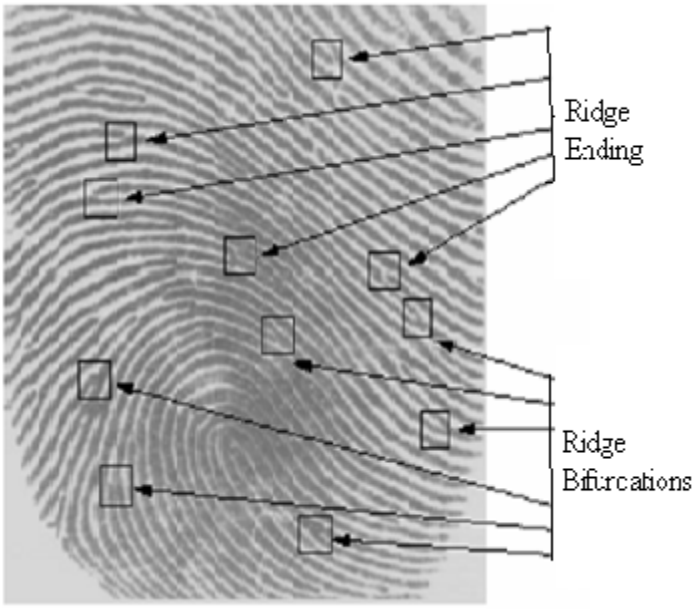


Figure 2.6 Ridge ending & Bifurcation

Many known algorithms have been developed for minutiae extraction based on orientation and gradients of the orientation fields of the ridges [3]. The building blocks of a fingerprint recognition system are:

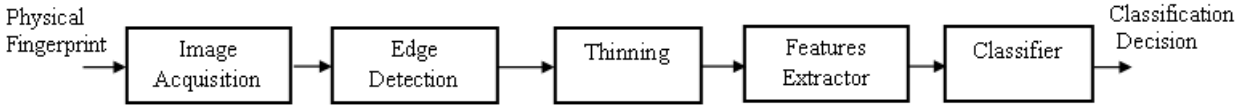


Figure 2.7 Fingerprint recognition system

a) Image Acquisition: A number of methods are used to acquire fingerprints. Among them, the inked impression method remains the most popular one. Inkless fingerprint scanners are also present eliminating the intermediate digitization process. Fingerprint quality is very important since it affects directly the minutiae extraction algorithm. Two types of degradation usually affect fingerprint images:

1) the ridge lines are not strictly continuous since they sometimes include small breaks (gaps);

2) parallel ridge lines are not always well separated due to the presence of cluttering noise. The resolution of the scanned fingerprints must be 500 dpi while the size is 300x300.

b) Edge Detection: An edge is the boundary between two regions with relatively distinct gray level properties. The idea underlying most edge-detection techniques is on the computation of a local derivative operator such as 'Roberts', 'Prewitt' or 'Sobel' operators. In practice, the set of pixels obtained from the edge detection algorithm seldom characterizes a boundary completely

because of noise, breaks in the boundary and other effects that introduce spurious intensity discontinuities. Thus, edge detection algorithms typically are followed by linking and other boundary detection procedures designed to assemble edge pixels into meaningful boundaries.

c) Thinning: An important approach to representing the structural shape of a plane region is to reduce it to a graph. This reduction may be accomplished by obtaining the skeleton of the region via thinning (also called skeletonizing) algorithm. The thinning algorithm while deleting unwanted edge points should not:

- Remove end points.
- Break connectedness
- Cause excessive erosion of the region

d) Feature Extraction: Extraction of appropriate features is one of the most important tasks for a recognition system. The feature extraction method used in [1] will be explained below. A multilayer perceptron (MLP) of three layers is trained to detect the minutiae in the thinned fingerprint image of size 300x300. The first layer of the network has nine neurons associated with the components of the input vector. The hidden layer has five neurons and the output layer has one neuron. The network is trained to output a "1" when the input window in centered on a minutiae and a "0" when it is not.

The networking will be trained using:

- The backpropagation algorithm with momentum and learning rate of 0.3.
- The Al-Alaoui backpropagation algorithm.

State the number of epochs needed for convergence as well as the training time for the two methods. Once the network is trained, the next step is to input the prototype fingerprint images to extract the minutiae.

e) Classifier: After scanning the entire fingerprint image, the resulting output is a binary image revealing the location of minutiae. In order to prevent any falsely reported output and select "significant" minutiae, two more rules are added to enhance the robustness of the algorithm:

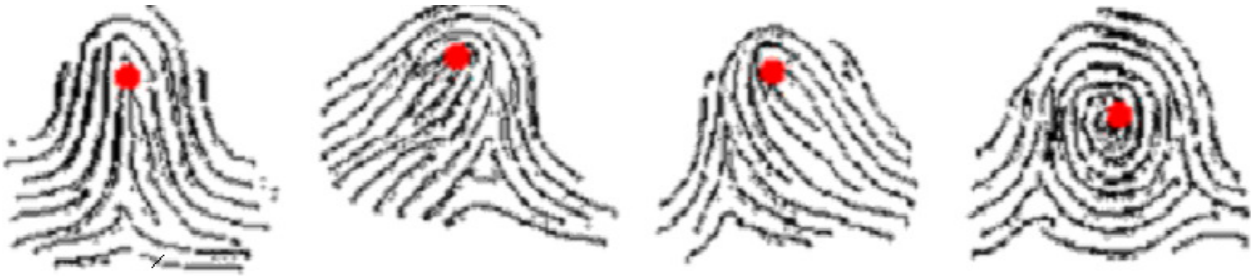
1) At those potential minutiae detected points, it can re-examine them by increasing the window size by 5x5 and scanning the output image.

2) If two or more minutiae are to close together (few pixels away) we ignore all of them.

To insure translation, rotation and scale-invariance, the following operations will be performed:

1. The Euclidean distance d(i) from each minutiae detected point to the center is calculated. The referencing of the distance data to the center point guarantees the property of positional invariance.
2. The data will be sorted in ascending order from d(0) to d(N), where N is the number of detected minutiae points, assuring rotational invariance.
3. The data is then normalized to unity by shortest distance d (0), i.e: dnorm(i) = d(0)/d(i); This will assure scale invariance property.

In the algorithm described above, the center of the fingerprint image was used to calculate the Euclidean distance between the center and the feature point. Usually, the center or reference point of the fingerprint image is what is called the "core" point. A core point, is located at the approximate center, is defined as the topmost point on the innermost upwardly curving ridgeline. Figure 2.8 shows some fingerprint patterns with the core point is marked. Many singularity points detection algorithms were investigated to locate core points, among them the famous "Poincaré" index method [25-26] and the one described in [27]. For simplicity we will assume that the core point is located at the center of the fingerprint image.



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

Figure 2.8 Core points on different fingerprint patterns. (a) tented arch, (b) right loop, (c) left loop, (d) whorl

After extracting the location of the minutiae for the prototype fingerprint images, the calculated distances will be stored in the database along with the ID or name of the person to whom each fingerprint belongs.

The last phase is the verification phase where testing fingerprint image:

1) is inputted to the system

2) minutiae are extracted

3) Minutiae matching: comparing the distances extracted minutiae to the one stored in the database

4) Identify the person

State the results obtained (i.e: recognition rate).

f) Efficiency of biometric system:

The effectiveness of a biometric system can be judged by following characteristics:

1. Performance: this refers to the achievable recognition accuracy, speed, robustness, the resource requirements to achieve the desired recognition accuracy and speed, as well as operational such as manual workers may have a large number of cuts and buries on their fingerprints, or environmental factors such as humidity, illumination etc., that affect the recognition accuracy and speed.
2. Scalability: This refers to the ability to encompass large number of individuals without a significant decrease in the performance.
3. Non-invasiveness: this refers to the ease with which the information can be captured.
4. From individuals, without damaging an individual's physical integrity and ideally without special preparations by/of individual.
5. Circumvention: This refers to the degree to which the system is resistant to spoofs or attacks. A practical biometric system should meet the specified recognition accuracy, speed, and resource requirements, be harmless to the users, be accepted by the intended population and be sufficiently robust to various fraudulent methods and attacks to the system.

g) Applications:

1. Government—Passports, national identification (ID) cards, voter cards, driver's licenses, social services, and so on.
2. Transportation—Airport security, boarding passes, and commercial driver's licenses.
3. Healthcare—Medical insurance cards, patient/employee identity cards.
4. Financial—Bankcards, ATM cards, credit cards, and debit cards.
5. Retail and gaming—Retail programs, such as check cashing, loyalty rewards and promotional cards, and gaming systems for access management and VIP programs
6. Security—Access control and identity verifications, including time and attendance.

7. Public justice and safety—Prison IDs, county probation offices' use for identification of parolees, county courthouses' use for ID systems.
8. Education—Student/teacher identity verification and access control. Biometrics are now being implemented in large-scale ID systems around the globe. Many new passport and national ID card systems use some type of biometric encoded in a bar code or smart chip.

2.7 Performance evaluation

The accuracy, and the response time constitutes the main performance parameters that must be reported in order to evaluate a certain biometric system. Although in the following discussion we will refer to fingerprint based biometric systems, the performance measures presented here are generally used in the evaluation of any other biometric system.

First of all, it is of importance to emphasize that the performance of a biometric system are random variables, and hence they cannot be measured, but can only be estimated based on empirical data. Unfortunately, the estimated performances are very data dependent being meaningful only for a specific collection of samples (database) captured in a specific test environment. An objective evaluation must rely on a large number of samples that are representative for the population of future users of the system, when such population can be identified. Several factors like the environmental conditions; the age, gender, occupation of the future users; etc, are of utmost importance in the data collection process. In the following we define the most important accuracy parameters used for system evaluation, assuming that a representative database of fingerprint samples has been collected.

A verification system performs a one-to-one comparison between test templates, and is reference template in order to decide either to accept, or reject the identity claimed by a certain person. There are two states of nature called respectively genuine, and impostor. The first one means that the test template and the reference template belong to the same finger (the person is a genuine user), and the second one means that the two templates belong to different fingers (the person is an impostor). The accept/reject decision of the system is based on the similarity between the two templates, which is quantified in a matching score. The higher the matching scores, the more likely it is that the person is a genuine user. Therefore, if a threshold (or decision criterion) is decided, the system will accept as genuine only those users whose test templates compared against the corresponding reference templates achieve a matching score higher than the

threshold.

In the context of biometrics the usual terms for the four possible outcomes o f the system are: False Accept, Correct Accept, False Reject, and Correct Reject. Obviously the first and the third outcomes are errors (called Type I and Type II error respectively), where as the other two outcomes are the ones sought. Manipulating the threshold value, the relative probabilities of the four outcomes can be adjusted such that to reflect their associated costs and benefits. These may be very different in different applications. In a forensic applications for instance, the cost of a False Reject error may exceed the cost of a False Accept error, whereas just the opposite may be true in a high security access application.

Given a collection of samples, the False Accept Rate (FAR), and the False Reject Rate (FRR) for a certain decision criterion can be estimated as follows

$$FAR = \frac{NFA}{NIE} \times 100\%, FRR = \frac{NFR}{NGE} \times 100\%$$

…… (2.1)

Where NFA and NFR are the number of impostor matching experiments that achieved matching scores above the threshold, and the numbers of genuine matching experiments that achieved matching scores below the threshold, respectively. NIE is the total number of impostor matching experiments, and NGE is the total number of genuine matching experiments

CHAPTER 3

LITERATURE SURVEY

This chapter presents the work done by other researcher related to fingerprint verification system, pores extraction and matching system. In this chapter description about all reference papers are summarized.

1.) Reliable extraction of features from poor quality prints is the most challenging problem faced in the area of fingerprint recognition. In this paper, author introduces a new approach for fingerprint image enhancement based on Short Time Fourier Transform (STFT) Analysis. STFT is a well known time-frequency analysis technique to analyze non-stationary signals. In this paper, the proposed analysis and enhancement algorithm simultaneously estimates several intrinsic properties of the fingerprint such as the foreground region mask, local ridge orientation and local frequency. They also objectively measure the effectiveness of the enhancement algorithm and show that it can improve the sensitivity and recognition accuracy of existing feature extraction and matching algorithms. They also present a new feature extraction algorithm based on chain code contour processing. Chain codes provide a loss-less representation of the fingerprint ridge contours along with providing us with a wide range of information about the contour such as curvature, direction, length etc. The algorithm has several advantages over the techniques proposed in literature such as increased computational efficiency, improved localization and higher sensitivity. They presented an objective evaluation of the feature extraction algorithm and show that it performs better than the existing approaches such as NIST MINDTCT algorithm. Finally they presented a novel minutia based fingerprint recognition algorithm that incorporates three new ideas. They also present CBFS (Coupled Breadth First Search), a new dual graph traversal algorithm for consolidating all the local neighborhood matches. They presented an experimental evaluation of the proposed approach and show that it performs better than the popular NIST BOZORTH3 matching algorithm [2].

2.) Qijun Zhao et al. [6] proposed an adaptive pore model for fingerprint pore extraction. Sweat pores have been recently employed for automated fingerprint recognition, in which the pores are usually extracted by using a computationally expensive skeletonization method or a unitary scale

isotropic pore model. In this paper, however, author shows that real pores are not always isotropic. To accurately and robustly extract pores, they propose an adaptive anisotropic pore model, whose parameters are adjusted adaptively according to the fingerprint ridge direction and period. The fingerprint image is partitioned into blocks and a local pore model is determined for each block. With the local pore model, a matched filter is used to extract the pores within each block. Experiments on a high resolution (1200dpi) fingerprint dataset are performed and the results demonstrate that the proposed pore model and pore extraction method can locate pores more accurately and robustly in comparison with other state-of- the-art pore extractors.

3.) Moheb R. et al. [7] proposed an approach to image extraction and accurate skin detection from web pages. This paper proposes a system to extract images from web pages and then detect the skin color regions of these images. As part of the proposed system, using BandObject control, they build a Tool bar named "Filter Tool Bar (FTB)" by modifying the Pavel Zolnikov implementation. In the proposed system, they introduce three new methods for extracting images from the web pages (after loading the web page by using the proposed FTB, before loading the web page physically from the local host, and before loading the web page from any server). These methods overcome the drawback of the regular expressions method for extracting images suggested by Ilan Assayag. The second part of the proposed system is concerned with the detection of the skin color regions of the extracted images. So, they studied two famous skin color detection techniques. The first technique is based on the RGB color space and the second technique is based on YUV and YIQ color spaces. They modified the second technique to overcome the failure of detecting complex image's background by using the saturation parameter to obtain an accurate skin detection results. The performance evaluation of the efficiency of the proposed system in extracting images before and after loading the web page from local host or any server in terms of the number of extracted images is presented. Finally, the results of comparing the two skin detection techniques in terms of the number of pixels detected are presented.

4.) Manvjeet Kaur et al. [8] proposed a fingerprint verification system using minutiae extraction technique. Most fingerprint recognition techniques are based on minutiae matching and have been well studied. However, this technology still suffers from problems associated with the handling of poor quality impressions. One problem besetting fingerprint matching is distortion.

Distortion changes both geometric position and orientation, and leads to difficulties in establishing a match among multiple impressions acquired from the same finger tip. Marking all the minutiae accurately as well as rejecting false minutiae is another issue still under research. Our work has combined many methods to build a minutia extractor and a minutia matcher. The combination of multiple methods comes from a wide investigation into research papers. Also some novel changes like segmentation using morphological operations, improved thinning, false minutiae removal methods, minutia marking with special considering the triple branch counting, minutia unification by decomposing a branch into three terminations, and matching in the unified x-y coordinate system after a two-step transformation are used in the work.

5.) Hoi Le et al. [9] proposed online fingerprint identification with a fast and distortion tolerant hashing method. National ID card, electronic commerce, and access to computer networks are some scenarios where reliable identification is a must. Existing authentication systems relying on knowledge-based approaches like passwords or token-based such as magnetic cards and passports contain serious security risks due to the vulnerability to engineering-social attacks and the easiness of sharing or compromising passwords and PINs. Biometrics such as fingerprint, face, eye retina, and voice offer a more reliable means for authentication. However, due to large biometric database and complicated biometric measures, it is difficult to design both an accurate and fast biometric recognition. Particularly, fast fingerprint indexing is one of the most challenging problems faced in fingerprint authentication system. In this paper, they present a specific contribution by introducing a new robust indexing scheme that is able not only to fasten the fingerprint recognition process but also improve the accuracy of the system.

6.) Ratha et al. [28] proposed an adaptive flow orientation based segmentation or binarization algorithm. In this approach the orientation field is computed to obtain the ridge directions at each point in the image. To segment the ridges, a 16x16 window oriented along the ridge direction is considered around each pixel. The projection sum along the ridge direction is computed. The centers of the ridges appear as peak points in the projection. The ridge skeleton thus obtained is smoothened by morphological operation. Finally minutiae are detected by locating end points and bifurcations in the thinned binary image.

7.) Anil Jain et al. [10] proposed a Pores and Ridges: Fingerprint Matching Using Level 3 Features. Fingerprint friction ridge details are generally described in a hierarchical order at three

levels, namely, Level 1 (pattern), Level 2 (minutiae points) and Level 3 (pores and ridge shape). Although high resolution sensors (~1000dpi) have become commercially available and have made it possible to reliably extract Level 3 features, most Automated Fingerprint Identification Systems (AFIS) employ only Level 1 and Level 2 features. As a result, increasing the scan resolution does not provide any matching performance improvement [17]. They develop a matcher that utilizes Level 3 features, including pores and ridge contours, for 1000dpi fingerprint matching. Level 3 features are automatically extracted using wavelet transform and Gabor filters and are locally matched using the ICP algorithm. Our experiments on a median-sized database show that Level 3 features carry significant discriminatory information. EER values are reduced (relatively ~20%) when Level 3 features are employed in combination with Level 1 and 2 features.

8.) Mayank Vatsa et al. [11] proposed an combining pores and ridges with minutiae for improved fingerprint verification. This paper presents a fast fingerprint verification algorithm using level-2 minutiae and level-3 pore and ridge features. The proposed algorithm uses a two-stage process to register fingerprint images. In the first stage, Taylor series based image transformation is used to perform coarse registration, while in the second stage, thin plate spline transformation is used for fine registration. A fast feature extraction algorithm is proposed using the Mumford–Shah functional curve evolution to efficiently segment contours and extracts the intricate level-3 pore and ridge features. Further, Delaunay triangulation based fusion algorithm is proposed to combine level-2 and level-3 information that provides structural stability and robustness to small changes caused due to extraneous noise or non-linear deformation during image capture. They defines eight quantitative measures using level-2 and level-3 topological characteristics to form a feature super vector. A 2n-support vector machine performs the final classification of genuine or impostor cases using the feature super vectors. Experimental results and statistical evaluation show that the feature super vector yields discriminatory information and higher accuracy compared to existing recognition and fusion algorithms.

9.) Umut Uludaga et al. [12] proposed a Biometric template selection and update: a case study in fingerprints. Sweat pores have been recently employed for automated fingerprint recognition, in which the pores are usually extracted by using a computationally expensive skeletonization method or a unitary scale isotropic pore model. In this paper, however, real pores are not always

isotropic. To accurately and robustly extract pores, they propose an adaptive anisotropic pore model, whose parameters are adjusted adaptively according to the fingerprint ridge direction and period. The fingerprint image is partitioned into blocks and a local pore model is determined for each block. With the local pore model, a matched filter is used to extract the pores within each block. Experiments on a high resolution (1200dpi) fingerprint dataset are performed and the results demonstrate that the proposed pore model and pore extraction method can locate pores more accurately and robustly in comparison with other state-of-the-art pore extractors.

10.) Coetzee and Botha [19] proposed a binarization technique based on the use of edges extracted using Marr-Hilderith operator. The resulting edge image is used in conjunction with the original gray scale image to obtain the binarized image. This is based on the recursive approach of line following and line thinning. Two adaptive windows, the edge window and the gray-scale window are used in each step of the recursive process. To begin with, the pixel with the lowest gray-scale value is chosen and a window is centered on it. The boundary of the window is then examined to determine the next position of the window. The window is successively position to trace the ridge boundary and the recursive process terminates when all the ridge pixels have been followed to their respective ends.

11.) O'Gorman et al. [31] proposed the use of contextual filters for fingerprint image enhancement for the first time. They used an anisotropic smoothening kernel whose major axis is oriented parallel to the ridges. For efficiency, they pre-compute the filter in 16 directions. The net result of the filter is that it increases contrast in a direction perpendicular to the ridges while performing smoothening in the direction of the ridges. Recently, Greenberg et al. [32] proposed the use of an anisotropic filter that is based on structure adaptive filtering.

12.) Ruud M. Bolle et al. [33] proposed the evaluation techniques for biometrics-based authentication systems (FRR). Biometrics-based authentication is becoming popular because of increasing ease-of-use and reliability. Performance evaluation of such systems is an important issue. They endeavor to address two aspects of performance evaluation that have been conventionally neglected. First, the "difficulty" of the data that is used in a study influences the evaluation results. They propose some measures to characterize the data set so that the performance of a given system on different data sets can be compared. Second, conventional studies often have reported the false reject and false accept rates in the form of match score

distributions. However, no confidence intervals are computed for these distributions, hence no indication of the significance of the estimates is given. In this paper, they compare the parametric and nonparametric (bootstrap) methods for measuring confidence intervals. They give special attention to false reject rate estimates.

13.) Wang Yuan et al. [34] proposed a real time fingerprint recognition system based on novel fingerprint matching strategy. In this paper they present a real time fingerprint recognition system based on a novel fingerprint minutiae matching algorithm. The system is developed to be applicable to today's embedded systems for fingerprint authentication, in which small area sensors are employed. The system is comprised of fingerprint enhancement and quality control, fingerprint feature extraction, fingerprint matching using a novel matching algorithm, and connection with other identification system. Here they describe their way to design a more reliable and fast fingerprint recognition system which is based on today's embedded systems in which small area fingerprint sensors are used. Experiment on FVC database show our system has a better performance than compared. And for the image enhancement and matching techniques they use high efficiency, it can also give a real time identification result with high reliability.

14.) Wei Cui et al. [35] proposed the research of edge detection algorithm for fingerprint images. This paper introduces some edge detection operators and compares their characteristics and performances. At last the experiment show that each algorithm has its advantages and disadvantages, and the suitable algorithm should be selected according the characteristic of the images detected, so that it can perform perfectly. The Canny Operator is not susceptible to the noise interference; it can detect the real weak edge. The advantage is that it uses two different thresholds to detect the strong edge and the weak edge, and the weak edge will be include in the output image only when the weak edge is connected to the strong edge. The Sobel Operator has a good performance on the images with gray gradient and high noise, but the location of edges is not very accurate, the edges of the image have more than one pixel. The Binary Image Edge Detection Algorithm is simple, but it can detect the edge of the image accurately, and the processed images are not need to be thinned, it particularly adapts to process various binary images with no noise. So each algorithm has its advantages and disadvantages, and the suitable algorithm should be selected according to the characters of the images been detected, then it can performance perfectly.

15.) Shunshan li et al. [36] proposed the Image Enhancement Method for Fingerprint Recognition System. In this paper fingerprint image enhancement method, a refined Gabor filter, is presented. This enhancement method can connect the ridge breaks, ensures the maximal gray values located at the ridge center and has the ability to compensate for the nonlinear deformations. it includes ridge orientation estimation, a Gabor filter processing and a refined Gabor filter processing. The first Gabor filter reduces the noise, provides more accurate distance between the two ridges for the next filter and gets a rough ridge orientation map while the refined Gabor filter with the adjustment parameters significantly enhances the ridge, connects the ridge breaks and ensures the maximal gray values of the image being located at the ridge center. In addition, the algorithm has the ability to compensate for the nonlinear deformations. Furthermore, this method does not result in any spurious ridge structure, which avoids undesired side effects for the subsequent processing and provides a reliable fingerprint image processing for Fingerprint Recognition System. In a word, a refined Gabor filter is applied in fingerprint image processing, then a good quality fingerprint image is achieved, and the performance of Fingerprint Recognition System has been improved.

16.) S. Mil'shtein et al. [37] proposed a fingerprint recognition algorithm for partial and full fingerprints. In this study, they propose two new algorithms. The first algorithm, called the Spaced Frequency Transformation Algorithm (SFTA), is based on taking the Fast Fourier Transform of the images. The second algorithm, called the Line Scan Algorithm (LSA), was developed to compare partial fingerprints and reduce the time taken to compare full fingerprints. A combination of SFTA and LSA provides a very efficient recognition technique. The most notable advantages of these algorithms are the high accuracy in the case of partial fingerprints. At this time, the major drawback of developed algorithms is lack of pre-classification of examined fingers. Thus, they use minutiae classification scheme to reduce the reference base for given tested finger. When the reference base had shrunk, they apply the LSA and SFTA.

17.) Another paper proposed a novel approaches for minutiae filtering in fingerprint images. Existing structural approaches for minutiae filtering use heuristics and adhoc rules to eliminate such false positives, where as gray level approach is based on using raw pixel values and a super-vised classifier such as neural networks. They proposed two new techniques for minutiae verification based on non-trivial gray level features. The proposed features intuitively represent

the structural properties of the minutiae neighborhood leading to better classification. They use directionally selective steerable wedge filters to differentiate between minutiae and non-minutiae neighborhoods with reasonable accuracy. They also propose a second technique based on Gabor expansions that result in even better discrimination. They present an objective evaluation of both the algorithms. Apart from minutiae verification, the feature description can also be used for minutiae detection and minutiae quality assessment.

18.) Arun Ross et al. [38] proposed biometric template selection and update: a case study in fingerprints. In this paper they propose two methods to perform automatic template selection where the goal is to select prototype fingerprint templates for a finger from a given set of fingerprint impressions. The first method, called DEND, employs a clustering strategy to choose a template set that best represents the intra-class variations, while the second method, called MDIST, selects templates that exhibit maximum similarity with the rest of the impressions. Matching results on a database of 50 different fingers, with 200 impressions per finger, indicate that a systematic template selection procedure as presented here results in better performance than random template selection. The proposed methods have also been utilized to perform automatic template update.

19.) Deepak Kumar Karna et al. [39] proposed normalized cross-correlation based fingerprint matching. To perform fingerprint matching based on the number of corresponding minutia pairings, has been in use for quite some time. But this technique is not very efficient for recognizing the low quality fingerprints. To overcome this problem, some researchers suggest the correlation technique which provides better result. Use of correlation-based methods is increasing day-by-day in the field of biometrics as it provides better results. In this paper, they propose normalized cross-correlation technique for fingerprint matching to minimize error rate as well as reduce the computational effort than the minutiae matching method. The EER (Equal Error Rate) obtained from result till now with minutiae matching method is 3%, while that obtained for the method proposed in this paper is approx 2% for all types of fingerprints in combined form.

20.) Asker M. Bazen et al. [40] proposed a correlation-based fingerprint verification system. In this paper, a correlation-based fingerprint verification system is presented. Unlike the traditional minutiae-based systems, this system directly uses the richer gray-scale information of the

fingerprints. The correlation-based fingerprint verification system first selects appropriate templates in the primary fingerprint, uses template matching to locate them in the secondary print, and compares the template positions of both fingerprints. Unlike minutiae-based systems, the correlation-based fingerprint verification system is capable of dealing with bad-quality images from which no minutiae can be extracted reliably and with fingerprints that suffer from non-uniform shape distortions. Experiments have shown that the performance of this system at the moment is comparable to the performance of many other fingerprint verification systems.

21.) David G. Lowe [41] proposed an approach to distinctive image features from scale-invariant keypoints. This paper presents a method for extracting distinctive invariant features from images that can be used to perform reliable matching between different views of an object or scene. The features are invariant to image scale and rotation, and are shown to provide robust matching across a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. The features are highly distinctive, in the sense that a single feature can be correctly matched with high probability against a large database of features from many images. This paper also describes an approach to using these features for object recognition. The recognition proceeds by matching individual features to a database of features from known objects using a fast nearest-neighbor algorithm, followed by a Hough transformation to identify clusters belonging to a single object, and finally performing verification through least-squares solution for consistent pores parameters. This approach to recognition can robustly identify objects among clutter and occlusion while achieving near real-time performance.

22.) Lin Zhang et al [43] proposed a new authentication system namely online finger-knuckle-print verification for personal authentication. Biometric based personal authentication is an effective method for automatically recognizing, with a high confidence, a person's identity. By observing that the texture pattern produced by bending the finger knuckle is highly distinctive, in this paper they present a new biometric authentication system using finger-knuckle-print (FKP) imaging. A specific data acquisition device is constructed to capture the FKP images, and then an efficient FKP recognition algorithm is presented to process the acquired data in real time. The local convex direction map of the FKP image is extracted based on which a local coordinate system is established to align the images and a region of interest is cropped for feature extraction. For matching two FKPs, a feature extraction scheme, which combines orientation and

magnitude information extracted by Gabor filtering, is proposed. An FKP database, which consists of 7920 images from 660 different fingers, is established to verify the efficiency of the proposed system and promising results are obtained. Compared with the other existing finger-back surface based biometric systems, the proposed FKP system achieves much higher recognition rate and it works in real time. It provides a practical solution to finger-back surface based biometric systems and has great potentials for commercial applications.

CHAPTER 4

FINGERPRINT LEVEL 3 FEATURES

This chapter describes general concepts of fingerprint level 3 features

4.1 Fingerprint Level 3 Features:

A good fingerprint representation should have the following two properties: Saliency and Suitability. Saliency means that representation should contain information about the fingerprint. Suitability means that the representation can be easily extracted, stored in compact fashion, be useful for matching. Saliency and Suitability are generally correlated. Image based representations constituted by raw pixel intensity information, are prevalent among the recognition system using optical matching and correlation based matching. However, the utility of these systems may be limited due to factor such as brightness variations, image quality variations, scars, and large global distortions present in the fingerprint image.

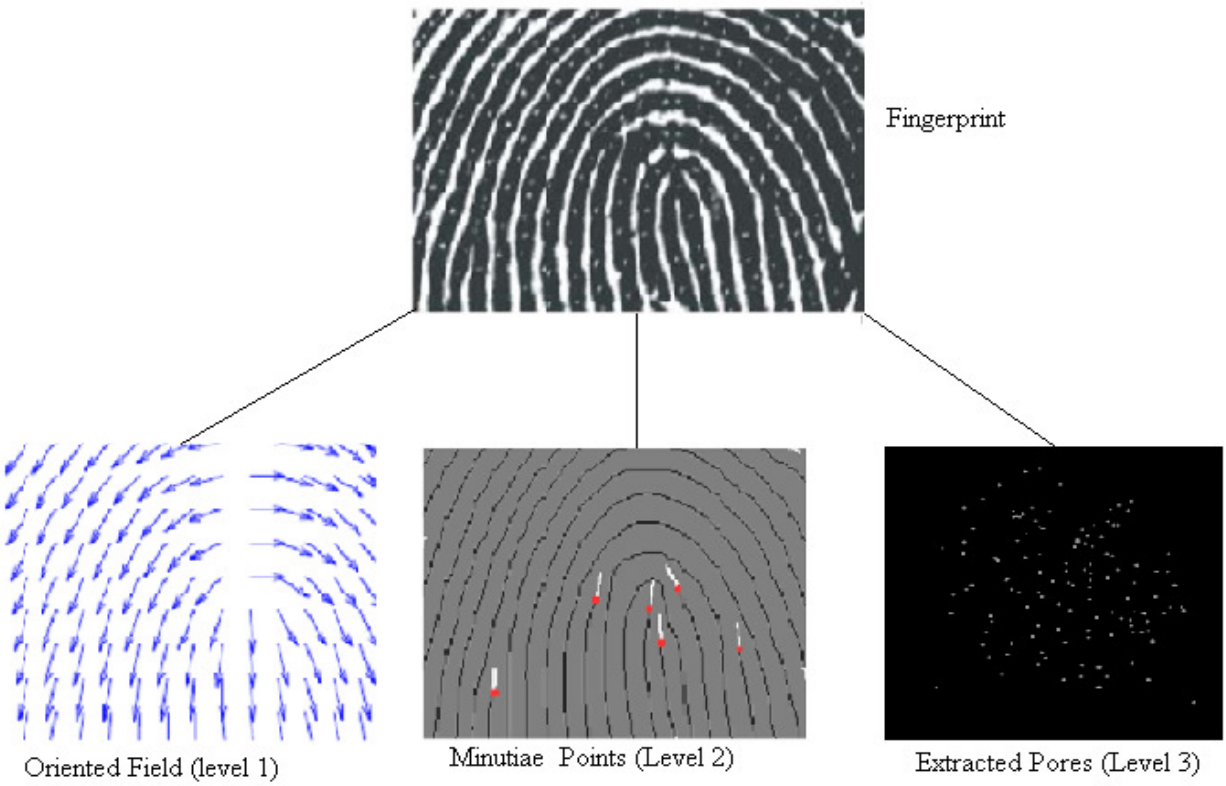


Figure 4.1 Different Level extracted from fingerprint

The types of information that can be collected from a fingerprint's friction ridge impression can be categorized as Level 1, Level 2, or Level 3 features as shown in Figure 4.1. At the global level, the fingerprint pattern exhibits one or more regions where the ridge lines assume distinctive shapes characterized by high curvature, frequent termination, etc. These regions are broadly classified into arch, loop, and whorl as already discussed in chapter 2. The arch, loop and whorl can further be classified into various subcategories. Level 1 feature comprises these global patterns and morphological information. They alone do not contain sufficient information to uniquely identify fingerprints but are used for broad classification of fingerprints.

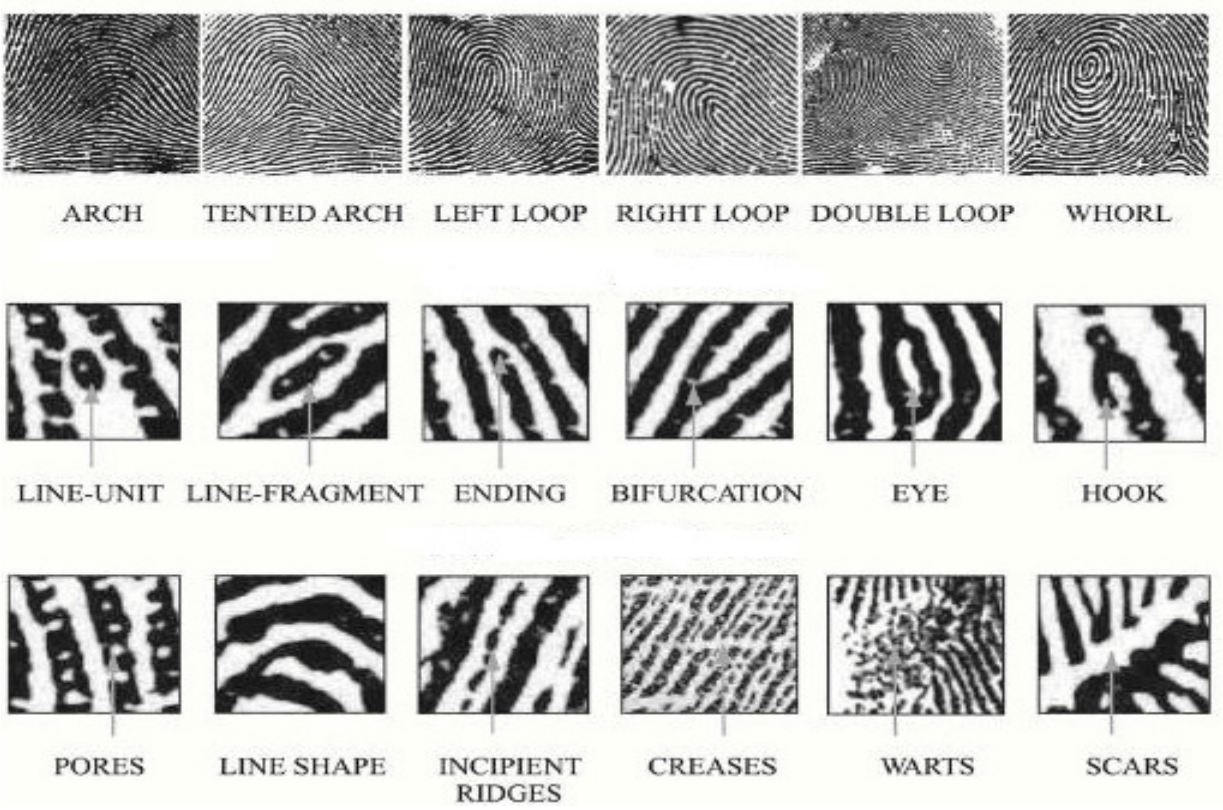


Figure 4.2 Different fingerprint features at different level.

Level 2 features or minutiae refer to the various ways that the ridges can be discontinuous. These are essentially Galton characteristics, namely ridge endings and ridge bifurcations. A ridge ending is defined as the ridge point where a ridge ends abruptly. A bifurcation is defined as the ridge point where a ridge bifurcates into two ridges. Minutiae are the most prominent features, generally stable and robust to fingerprint impression conditions. The distribution of minutiae in a fingerprint is considered unique and most of the automated matchers use this property to uniquely identify fingerprints. Uniqueness of fingerprint based on minutia points has been identifying

fingerprints. Uniqueness of fingerprint based on minutia points has been quantified by Galton [22] Statistical analysis has shown that Level 2 features, have sufficient discriminating power to establish the individuality of fingerprints pores should be sufficient to determine the identity of an individual.

Level 3 features refer to pores and contour ridges. Pores are extremely fine detail which is use in level 3 fingerprint matching technique. These are essentially the sweat pores and ridge contours. Pores are the openings of the sweat glands and they are distributed along the ridges. Studies that density of pores on a ridge varies from 23 to 45 pores per inch and 20 to 40 pores should be sufficient to determine the identity of an individual. A pore can be either open or closed, based on its perspiration activity. A closed pore is entirely enclosed by a ridge, while an open pore intersects with the valley lying between two ridges as shown in Figure 4.3.

The pore information (position, number and shape) are considered to be permanent, immutable and highly distinctive but very few automatic matching techniques use pores since their reliable extraction requires high resolution and good quality fingerprint images. Ridge contours contain valuable Level 3 information including ridge width and edge shape. Various shapes on the friction ridge edges can be classified into eight categories, namely, straight, convex, peak, table, pocket, concave, angle, and others as shown in Figure 4.2 and relative position of ridge edges are considered as permanent and unique.

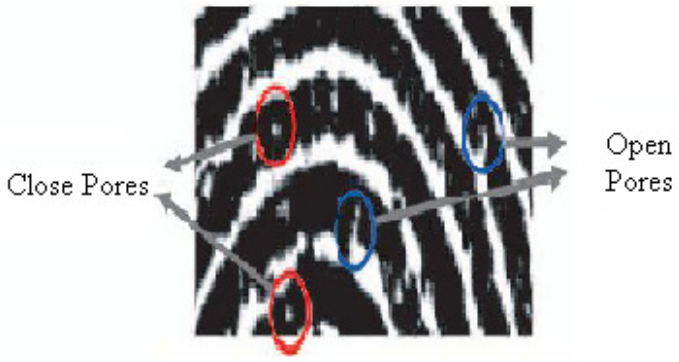


Figure 4.3 Open and Close Pores

4.2 Pores Extraction Technique

Pores are extremely fine detail which is use in level 3 fingerprint matching technique. Pores are directly extracted from the preprocessed image. The pore extraction can be classified in to two classes: first class of algorithm extract the pores by tracing fingerprint skeletons, second

algorithm extract the pores directly from gray scale image. Stosz ez al. [7] and kryszczuk [8] have proposed skeletonization process for pore extraction. Skeletonization based approach is reliable for extracting pores in high quality image. As the image resolution decreases or the skin condition is not favorable, the method does not give the reliable results. In [9] Jain et al proposed a pore extraction technique directly from gray scale image. The pores are distributed over ridges and using orientation detail can provide additional information for matching. A recent study [11] by the international Biometric Group has proposed new approach for pore extraction which utilizes orientation information of pores along with the location information. The opening in pore extraction process is the valuation of the ridge orientation. The local ridge orientation is determined by the least square estimate method [13]. This data is utilized later in the representation of pores.

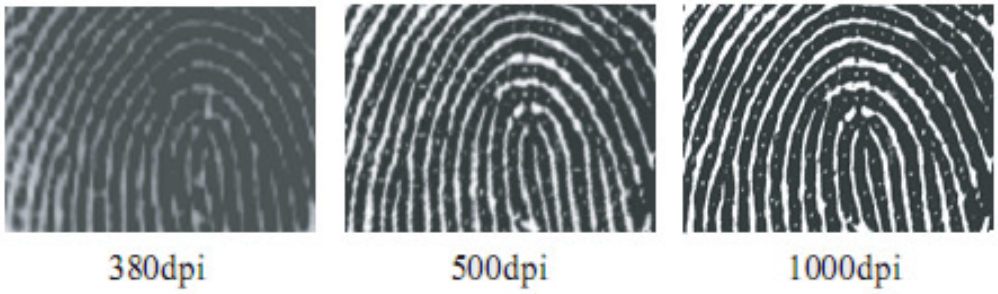


Figure 4.4 Fingerprint Features. (a) A partial fingerprint image captured at various resolutions (380dpi, 500dpi, and 1000dpi) using Identix 200DFR and CrossMatch ID1000 sensors. (b) Features extracted at different levels from the 1000dpi fingerprint in (a).

Statistical analysis has shown that Level 1 features, or fingerprint pattern, though not unique, are useful for classification purpose, while Level 2 features, or points, have sufficient discriminating power to establish the individuality of fingerprints [17, 18]. FBI has set the standard for fingerprint resolution to be 500dpi for forensic applications in order to reliably extract Level 2 features. However, human examiners perform not only quantitative (Level 2) but also qualitative (Level 3) examination since Level 3 features are also permanent, immutable and unique [6]. As stated by latent print examiner Ashbaugh, “It is not the points, but what’s in between the points that matters” [7]. With the availability of high resolution sensors (≥ 1000dpi), richer features can be extracted as shown in Figure 4.4. Hence, it is desirable to investigate performance improvement by introducing Level 3 features in fingerprint matching. Use of Level 3 features in fingerprint matching was studied by Kryszczuk [15] and, Roddy and Stoz [23]. They focused on

pore-based Level 3 matching using fingerprint fragments, but the alignment of the template and query fragments is either manually determined or pre-defined. Unlike these studies, our system uses the entire fingerprint for matching.

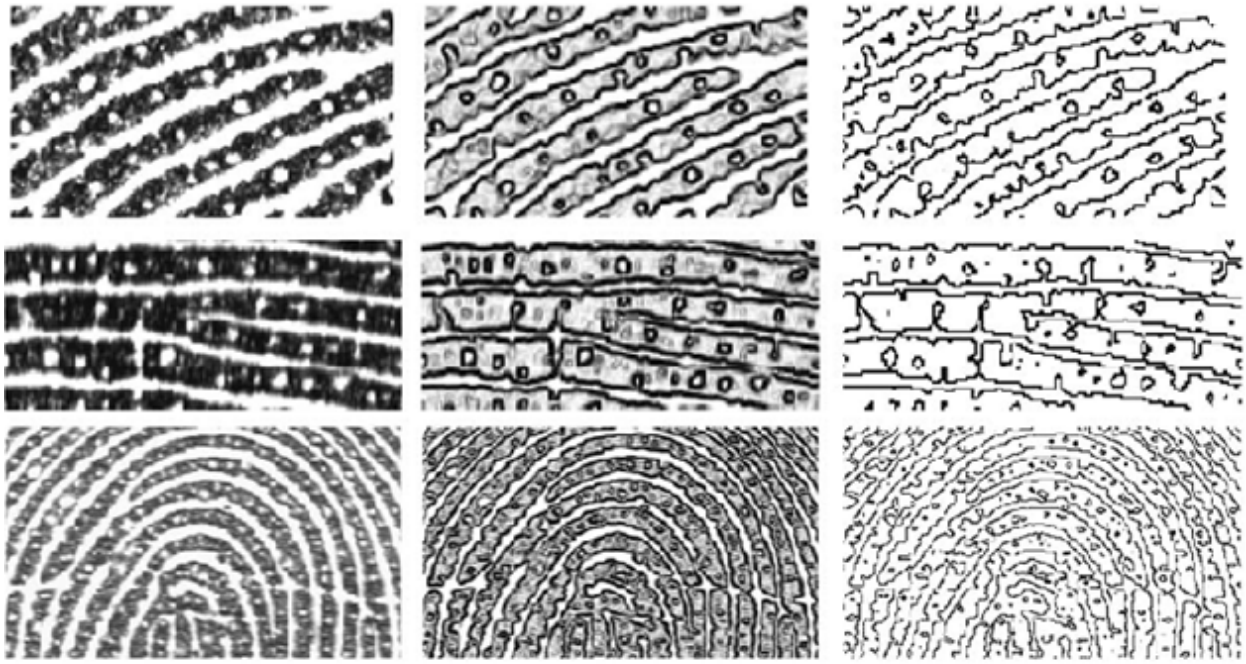


Figure 4.5 Images illustrating the intermediate steps of the level-3 feature extraction algorithm: (a) input fingerprint image, (b) stopping term f computed using Eq. (3), and (c) extracted fingerprint contour.

The level-3 feature extraction algorithm uses curve evolution with the fast implementation of Mum-ford–Shah functional [21, 22]. Mumford–Shah curve evolution efficiently segments the contours present in the image. In this approach, feature boundaries are detected by evolving a curve and minimizing energy based segmentation model as defined in the following equation [21]:

$$\text{Energy}(C, c_1, c_2)=\alpha \iint_{\pi} \varphi\|\bar{C}\| \, dxdy + \beta \iint_{in(C)} |I(x, y) - c_1|^2 \, dxdy + \gamma \iint_{out(C)} |I(x, y)-c_2|^2 \, dxdy \quad \text{..... (4.1)}$$

where $\bar{C}$ is the evolution curve such that $\bar{C} = \{(x,y): \varphi(x,y) = 0\}$, C is the is the evolution curve parameter, φ is the weighting function or stopping term, Ω represent the image domain, *I(x, y)* represent fingerprint image, c_1 and c_2 are the average value of pixels inside and outside $\bar{C}$, respectively, and α, β, and γ are positive constant such that α + β + γ=1 and α<β <γ. Further,

Chan and Vese [22] parameterize the energy equation (Eq. 4.1) by the artificial time t≥0 and deduce the associated Euler-Lagrange equation and that leads to the following active contour model,

$$\bar{\psi}_t = \alpha\, \phi(\bar{v}+\varepsilon_k)|\nabla\bar{\psi}| + \nabla\phi\nabla\bar{\psi} + \beta\, \delta(1-c_1)^2 + \gamma\, \delta\, \bar{\psi}(1-c_2)^2 \quad \text{…….. (4.2)}$$

Where $\bar{v}$ the advection is term and ε_k is the curvature based smoothing term. ∇ is the gradient and δ=0.5/(π(x^2+0.25)). The stopping term ϕ is set to

$$\phi = \frac{1}{1-(|\nabla I|)^2} \quad \text{…….. (4.3)}$$

This gradient based stopping term ensures that at the strongest gradient, the speed of the curve evolution becomes zero and therefore it stops at the edge of the image.

Initial contour is initialized as a grid with 750 blocks over the fingerprint image and the boundary of each feature is computed. The size of grid is chosen to balance the time required for curve evolution and correct extraction of all the features. Figure 4.5 shows examples of feature extraction from fingerprint images. This image shows that due to the stopping term (Figure 4.5b), the noise present in the fingerprint image has very little effect on contour extraction.

Once the contour extraction algorithm provides final fingerprint contour ψ, it is scanned using the standard contour tracing technique [11] from top to bottom and left to right consecutively. Tracing and classification of level-3 pore and ridge features are performed based on the standards defined by the ANSI/ NIST committee for extended fingerprint feature set (CDEFFS) [42]. During tracing, the algorithm classifies the contour information into pores and ridges:

(1) A blob of size greater than 2 pixels and less than 40 pixels is classified as a pore. Therefore, noisy contours, which are sometimes wrongly extracted, are not included in the feature set. A pore is approximated with a circle and the center is used as the pore feature [11].

(2) An edge of a ridge is defined as the ridge contour. Each row of the ridge feature represents x; y coordinates of the pixel and direction of the contour at that pixel [11].

CHAPTER 5

SIFT ALGORITHM

Scale Invariant Features Transform (SIFT) is an algorithm in computer vision to detect and describe local features in images. The algorithm was published by David Lowe [45] in 1999. Applications include object recognition, robotic mapping and navigation, image stitching, 3D modeling, gesture recognition, video tracking, and match moving. SIFT technique describes image features that have many properties that make them suitable for matching differing images of an object or scene. The features are invariant to image scaling and rotation, and partially invariant to change in illumination and 3D camera viewpoint. They are well localized in both the spatial and frequency domains, reducing the probability of disruption by occlusion, clutter, or noise. Large numbers of features can be extracted from typical images with efficient algorithms. In addition, the features are highly distinctive, which allows a single feature to be correctly matched with high probability against a large database of features, providing a basis for object and scene recognition. The cost of extracting these features is minimized by taking a cascade filtering approach, in which the more expensive operations are applied only at locations that pass an initial test.

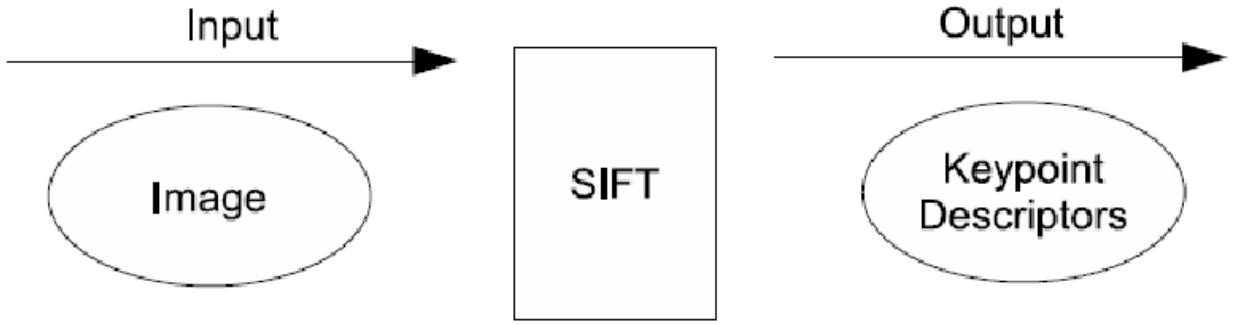


Figure 5.1 SIFT takes as input an image, and generates a set of keypoint descriptors.

Following are the major stages of computation used to generate the set of image features:

1 Scale-space extreme detection: The first stage of computation searches over all scales and image locations. It is implemented efficiently by using a difference-of-Gaussian function to identify potential interest points that are invariant to scale and orientation.

2 Keypoint localization: At each candidate location, a detailed model is fit to determine location and scale. Keypoints are selected based on measures of their stability.

3 Orientation assignments: One or more orientations are assigned to each keypoint location based on local image gradient directions. All future operations are performed on image data that has been transformed relative to the assigned orientation, scale, and location for each feature, thereby providing invariance to these transformations.

4 Keypoint descriptor: The local image gradients are measured at the selected scale in the region around each keypoint. These are transformed into a representation that allows for significant levels of local shape distortion and change in illumination.

This approach has been named the Scale Invariant Feature Transform (SIFT), as it transforms image data into scale-invariant coordinates relative to local features. An important aspect of this approach is that it generates large numbers of features that densely cover the image over the full range of scales and locations. A typical image of size 500x500 pixels will give rise to about 2000 stable features (although this number depends on both image content and choices for various parameters). The quantity of features is particularly important for object recognition, where the ability to detect small objects in cluttered backgrounds requires that at least 3 features be correctly matched from each object for reliable Identification.

The keypoint descriptors are highly idiosyncratic, which allows a single feature to find its correct match with good probability in a large database of features. However, in a cluttered image, many features from the background will not have any correct match in the database, giving rise to many false matches in addition to the correct ones. The correct matches can be filtered from the full set of matches by identifying subsets of keypoints that agree on the object and its location, scale, and orientation in the new image. The probability that several features will agree on these parameters by chance is much lower than the probability that any individual feature match will be in error. The determination of these consistent clusters can be performed rapidly by using an efficient hash table implementation of the generalized Hough transform.

Each cluster of 3 or more features that agree on an object and its pose is then subject to further detailed verification. First, a least-squared estimate is made for an affine approximation to the object pose. Any other image features consistent with this pose are identified, and outliers are discarded. Finally, a detailed computation is made of the probability that a particular set of features indicates the presence of an object, given the accuracy of fit and number of probable

false matches. Object matches that pass all these tests can be identified as correct with high confidence.

While Lowes [44] suggests several ways to select keypoints from the candidate keypoints, there is no precise characterization of keypoints in terms of human perception of an image. Furthermore, the experience with the results of implementations of SIFT supports the view that there is no well defined connection between keypoints and the perceptual characteristics of an image. It is possible to obtain some analytical results for a limited class of images, as shown next.

5.1 Keypoints and keypoint Descriptors

As already mentioned, the SIFT algorithm produces keypoint descriptors. A keypoint is an image feature which is so distinct that image scaling, noise, or rotation does not, or rather should not, distort the keypoint. That is, given a keypoint in an image, if one scales the image to half the size, or double the size, the keypoint would still be identifiable. The same goes for image rotation and noise. If an image is, for example, rotated clockwise, the keypoint would still persist. A keypoint descriptor is a 128-dimensional vector that describes a keypoint. The reason for this high dimension is that each keypoint descriptor contains a lot of information about the point it describes. We shall in the next section have a closer look at what information the keypoint descriptors hold when we discuss the four phases of SIFT. To illustrate what a keypoint is, let us have a look at some images where keypoint descriptors have been detected.

5.2 Detection of scale-space extrema

As described in the introduction, there will be detection of keypoints using a cascade filtering approach that uses efficient algorithms to identify candidate locations that are then examined in further detail. The first stage of keypoint detection is to identify locations and scales that can be repeat ably assigned under differing views of the same object. Detecting locations that are invariant to scale change of the image can be accomplished by searching for stable features across all possible scales, using a continuous function of scale known as scale space (Witkin,1983).

It has been shown by Lowes [45] that under a variety of reasonable assumptions the only possible scale-space kernel is the Gaussian function. Therefore, the scale space of an image is defined as a

function, L(x, y, σ), that is produced from the convolution of a variable-scale Gaussian, G(x, y, σ), with an input image, I(x, y):

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad \text{…….. (5.1)}$$

where $*$ is the convolution operation in x and y, and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad \text{…….. (5.2)}$$

To efficiently detect stable keypoints location in scale space, some authors had proposed [45] using scale-space extrema in the difference-of-Gaussian function convolved i^{th} the image, D(x, y, σ), which can be computed from the difference of two nearby scales separated by a constant multiplicative factor k:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \quad \text{……....……. (5.3)}$$

There are a number of reasons for choosing this function. First, it is a particularly efficient function to compute, as the smoothed images, L, need to be computed in any case for scale space feature description, and D can therefore be computed by simple image subtraction.

In addition, the difference-of-Gaussian function provides a close approximation to the scale-normalized Laplacian of Gaussian, $\sigma^2\nabla^2 G$, as studied by Lindeberg. Lindeberg showed that the normalization of the Laplacian with the factor σ^2 is required for true scale invariance. In detailed experimental comparisons, Mikolajczyk (2002) found that the maxima and minima of $\sigma^2\nabla^2 G$, produce the most stable image features compared to a range of other possible image functions, such as the gradient, Hessian, or Harris corner function. The relationship between D and $\sigma^2\nabla^2 G$, can be understood from the heat diffusion equation (parameterized in terms of σ rather than the more usual $t = \sigma^2$):

$$\frac{\partial G}{\partial \sigma} = \sigma\nabla^2 G \quad \text{…….. (5.4)}$$

From this, it is seen that $\nabla^2 G$, can be computed from the finite difference approximation to $\partial G/\partial\sigma$, using the difference of nearby scales at kσ and σ:

$$\nabla^2 G = \frac{\partial G}{\partial \sigma} \approx \frac{G(x, y, k\sigma) - G(x, y, \sigma)}{k\sigma - \sigma}$$

…… (5.5)

and therefore,

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k - 1)\,\sigma^2 \nabla^2 G$$

…… (5.6)

This shows that when the difference-of-Gaussian function has scales differing by a constant factor it already incorporates the σ^2 scale normalization required for the scale-invariant

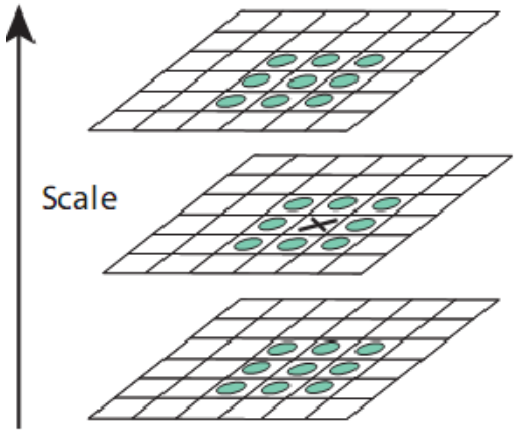


Figure 5.2 Maxima and minima of the difference-of-Gaussian images are detected by comparing a pixel (marked with X) to its 26 neighbors in 3x3 regions at the current and adjacent scales (marked with circles).

Laplacian. The factor (k − 1) in the equation is a constant over all scales and therefore does not influences extrema location. The approximation error will go to zero as k goes to 1, but in practice it has found that the approximation has almost no impact on the stability of extrema detection or localization for even significant differences in scale, such as k =√2.

An efficient approach to construction of D(x, y, σ) is shown in Figure 5.2. The initial image is incrementally convolved with Gaussians to produce images separated by a constant factor k in scale space, shown stacked in the left column. It can be choose to divide each octave of scale space (i.e., doubling of σ) into an integer number, s, of intervals, so k = 21/s. it must produce s + 3 images in the stack of blurred images for each octave, so that final extrema detection covers a complete octave. Adjacent image scales are subtracted to produce the difference-of-Gaussian images shown on the right. Once a complete octave has been processed, it resample the Gaussian

image that has twice the initial value of σ (it will be 2 images from the top of the stack) by taking every second pixel in each row and column. The accuracy of sampling relative to σ is no different than for the start of the previous octave, while computation is greatly reduced.

5.3 Local extrema detection

In order to detect the local maxima and minima of D(x, y, σ), each sample point is compared to its eight neighbors in the current image and nine neighbors in the scale above and below (Figure 5.3). It is selected only if it is larger than all of these neighbors or smaller than all of them. The cost of this check is reasonably low due to the fact that most sample points will be eliminated following the first few checks. An important issue is to determine the frequency of sampling in the image and scale domains that are needed to reliably detect the extrema. Unfortunately, it turns out that there is no minimum spacing of samples that will detect all extrema, as the extrema can be arbitrarily close together.

This can be seen by considering a white circle on a black background, which will have a single scale space maximum where the circular positive central region of the difference-of-Gaussian function matches the size and location of the circle. For a very elongated ellipse, there will be two maxima near each end of the ellipse. As the locations of maxima are a continuous function of the image, for some ellipse with intermediate elongation there will be a transition from a single maximum to two, with the maxima arbitrarily close to each other near the transition.

Therefore, it must settle for a solution that trades off efficiency with completeness. In fact, as might be expected and is confirmed by our experiments, extrema that are close together are quite unstable to small perturbations of the image. It can determine the best choices experimentally by studying a range of sampling frequencies and using those that provide the most reliable results under a realistic simulation of the matching task.

5.4 Frequency of sampling in scale

The experimental determination of sampling frequency that maximizes extrema stability is shown in Figures 3 and 4. These Figures (and most other simulations in this paper) are based on a matching task using a collection of 32 real images drawn from a diverse range, including outdoor scenes, human faces, aerial photographs, and industrial images (the image domain was found to have almost no influences on any of the results). Each image was then subject to a range

of transformations, including rotation, scaling, affine stretch, change in brightness and contrast, and addition of image noise. Because the changes were synthetic, it was possible to precisely predict where each feature in an original image should appear in the transformed image, allowing for measurement of correct repeatability and positional accuracy for each feature.

Simulation results used to examine the effect of varying the number of scales per octave at which the image function is sampled prior to extrema detection. In this case, each image was resembled following rotation by a random angle and scaling by a random amount between 0.2 of 0.9 times the original size. Keypoints from the reduced resolution image were matched against those from the original image so that the scales for all keypoints would be present in the matched image. In addition, 1% image noise was added, meaning that each pixel had a random number added from the uniform interval [-0.01, 0.01] where pixel values are in the range [0, 1] (equivalent to providing slightly less than 6 bits of accuracy for image pixels).

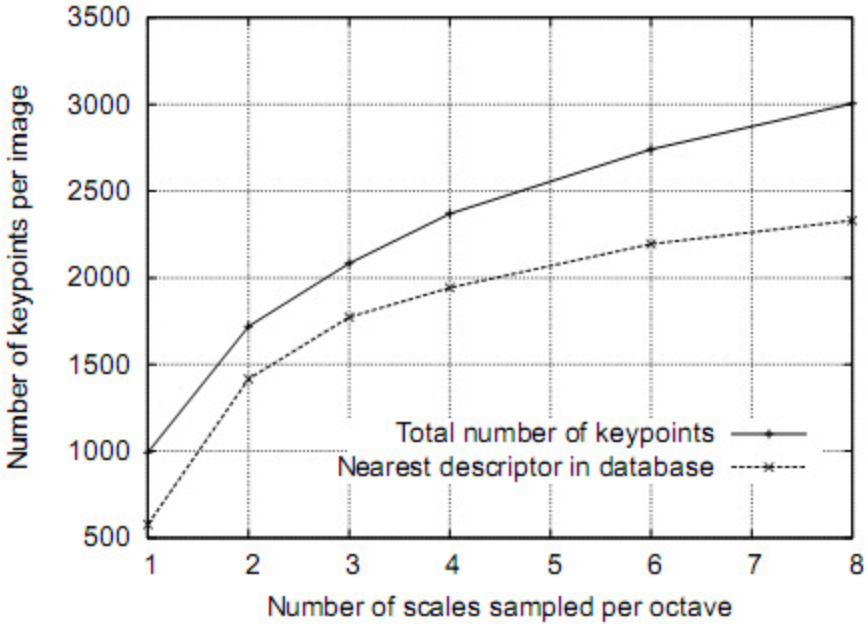


Figure 5.3 The graph shows the total number of keypoints detected in a typical image as a function of the number of scale samples.

The top line in the graph of Figure 5.3 shows the percent of keypoints that are detected at a matching location and scale in the transformed image. For all examples in this paper, it can be defines as a matching scale as being within a factor of $\sqrt{2}$ of the correct scale, and a matching location as being within σ pixels, where σ is the scale of the keypoint (defined from equation

(5.1) as the standard deviation of the smallest Gaussian used in the difference-of-Gaussian function). The lower line on this graph shows the number of keypoints that are correctly matched to a database of 40 keypoints using the nearest-neighbor matching procedure to be described in Section 5.10 (this shows that once the keypoint is repeatedly located, it is likely to be useful for recognition and matching tasks).

As Figure 5.3 graph shows, the highest repeatability is obtained when sampling 3 scales per octave. It might seem surprising that the repeatability does not continue to improve as more scales are sampled. The reason is that this results in many more local extrema being detected, but these extrema are on average less stable and therefore are less likely to be detected in the transformed image. This is shown by the graph in Figure 5.3, which shows the average number of keypoints detected and correctly matched in each image. The number of keypoints rises with increased sampling of scales and the total number of correct matches also rises. Since the success of object recognition often depends more on the quantity of correctly matched keypoints, as opposed to their percentage correct matching, for many applications it will be optimal to use a larger number of scale samples. However, the cost of computation also rises with this number, so for the experiments in this chapter it has to be chosen to use just 3 scale samples per octave.

To summarize, these experiments show that the scale-space difference-of-Gaussian function has a large number of extrema and that it would be very expensive to detect them all. Fortunately, we can detect the most stable and useful subset even with a coarse sampling of scales.

5.5 Frequency of sampling in the spatial domain

Just as it can be determining the frequency of sampling per octave of scale space, so there must determine the frequency of sampling in the image domain relative to the scale of smoothing. Given that extrema can be arbitrarily close together, there will be a similar trade-off between sampling frequency and rate of detection.

Therefore, to make full use of the input, the image can be expanded to create more sample points than were present in the original. So double the size of the input image using linear interpolation prior to building the first level of the pyramid. While the equivalent operation could have been performed effectively by using sets of subpixel-offset filters on the original image, but the image doubling leads to a more efficient implementation.

It can be assume that the original image has a blur of at least σ = 0.5 (the minimum needed to prevent significant aliasing), and that therefore the doubled image has σ = 1.0 relative to its new pixel spacing. This means that little additional smoothing is needed prior to creation of the first octave of scale space. The image doubling increases the number of stable keypoints by almost a factor of 4, but no significant further improvements was found with a larger expansion factor.

5.6 Accurate keypoint localization

Once a keypoint candidate has been found by comparing a pixel to its neighbors, the next step is to perform a detailed fit to the nearby data for location, scale, and ratio of principal curvatures. This information allows points to be rejected that have low contrast (and are therefore sensitive to noise) or are poorly localized along an edge.

The initial implementation of this approach [46] simply located keypoints at the location and scale of the central sample point. However, recently Brown [47] has developed a method for fitting a 3D quadratic function to the local sample points to determine the interpolated location of the maximum, and his experiments showed that this provides a substantial improvement to matching and stability. His approach uses the Taylor expansion (up to the quadratic terms) of the scale-space function, D(x, y, σ), shifted so that the origin is at the sample point:

$$D(x)=D+\frac{\delta D^{T}}{\delta x}x+\frac{1}{2}x^{T}\frac{\delta^{2}D}{\delta x^{2}}x$$

…… (5.7)

where D and its derivatives are evaluated at the sample point and x = (x, y, σ)T is the offset from this point. The location of the extremum, ˆx, is determined by taking the derivative of this function with respect to x and setting it to zero, giving:

$$\hat{x}=-\frac{\delta^{2}D^{-1}}{\partial x^{2}}\frac{\partial D}{\partial x}$$

…… (5.8)

As suggested by Brown, the Hessian and derivative of D are approximated by using differences of neighboring sample points. The resulting 3x3 linear system can be solved with minimal cost. If the offset ˆx is larger than 0.5 in any dimension, then it means that the extremum lies closer to a different sample point. In this case, the sample point is changed and the interpolation performed instead about that point. The final offset ˆx is added to the location of its sample point to get the interpolated estimate for the location of the extreme.

For stability, it is not sufficient to reject keypoints with low contrast. The difference-of-Gaussian function will have a strong response along edges, even if the location along the edge is poorly determined and therefore unstable to small amounts of noise.

A poorly defined peak in the difference-of-Gaussian function will have a large principal curvature across the edge but a small one in the perpendicular direction. The principal curvatures can be computed from a 2x2 Hessian matrix, H, computed at the location and scale of the keypoint:

$$H = \begin{matrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{matrix} \quad \text{..... (5.9)}$$

The derivatives are estimated by taking differences of neighboring sample points.

5.7 Orientation assignment

By assigning a consistent orientation to each keypoint based on local image properties, the keypoint descriptor can be represented relative to this orientation and therefore achieve invariance to image rotation. This approach contrasts with the orientation invariant descriptors of Schmid and Mohr (1997), in which each image property is based on a rotationally invariant measure. The disadvantage of that approach is that it limits the descriptors that can be used and discards image information by not requiring all measures to be based on a consistent rotation.

Following experimentation with a number of approaches to assigning a local orientation, the following approach was found to give the most stable results. The scale of the keypoint is used to select the Gaussian smoothed image, L, with the closest scale, so that all computations are performed in a scale-invariant manner. An orientation histogram is formed from the gradient orientations of sample points within a region around the keypoint. The orientation histogram has 36 bins covering the 360 degree range of orientations. Each sample added to the histogram is weighted by its gradient magnitude and by a Gaussian-weighted circular window with an σ that is 1.5 times that of the scale of the keypoint.

Peaks in the orientation histogram correspond to dominant directions of local gradients. The highest peak in the histogram is detected, and then any other local peak that is within 80% of the highest peak is used to also create a keypoint with that orientation. Therefore, for locations with multiple peaks of similar magnitude, there will be multiple keypoints created at the same location and scale but different orientations. Only about 15% of points are assigned multiple

orientations, but these contribute significantly to the stability of matching. Finally, a parabola is fit to the 3 histogram values closest to each peak to interpolate the peak position for better accuracy.

5.8 The local image descriptor

The previous operations have assigned an image location, scale, and orientation to each keypoint. These parameters impose a repeatable local 2D coordinate system in which to describe the local image region, and therefore provide invariance to these parameters. The next step is to compute a descriptor for the local image region that is highly distinctive yet is as invariant as possible to remaining variations, such as change in illumination or 3D viewpoint. One obvious approach would be to sample the local image intensities around the key-point at the appropriate scale, and to match these using a normalized correlation measure. However, simple correlation of image patches is highly sensitive to changes that cause mis-registration of samples, such as affine or 3D viewpoint change or non-rigid deformations. A better approach has been demonstrated by Edelman, Intrator, and Poggio (1997). Their proposed representation was based upon a model of biological vision, in particular of complex neurons in primary visual cortex. These complex neurons respond to a gradient at a particular orientation and spatial frequency, but the location of the gradient on the retina is allowed to shift over a small receptive field rather than being precisely localized.

Edelman et al. hypothesized that the function of these complex neurons was to allow for matching and recognition of 3D objects from a range of viewpoints. They have performed detailed experiments using 3D computer models of object and animal shapes which show that matching gradients while allowing for shifts in their position results in much better classification under 3D rotation. For example, recognition accuracy for 3D objects rotated in depth by 20 degrees increased from 35% for correlation of gradients to 94% using the complex cell model. Our implementation described below was inspired by this idea, but allows for positional shift using a different computational mechanism.

5.9 Descriptor representation

Figure 5.4 illustrates the computation of the keypoint descriptor. First the image gradient magnitudes and orientations are sampled around the keypoint location, using the scale of the keypoint to select the level of Gaussian blur for the image. In order to achieve orientation invariance, the coordinates of the descriptor and the gradient orientations are rotated relative to the keypoint orientation. These are illustrated with small arrows at each sample location on the left side of Figure 5.4

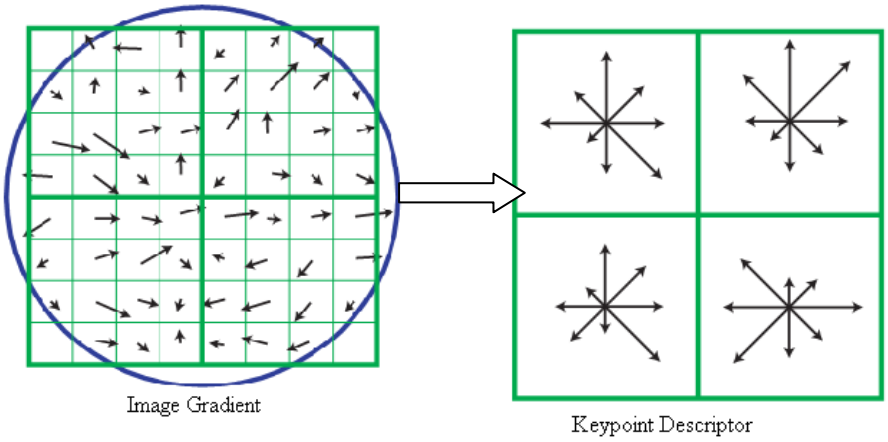


Figure 5.4 A keypoint descriptor is created by first computing the gradient magnitude and orientation at each image sample point in a region around the keypoint location, as shown on the left.

These are weighted by a Gaussian window, indicated by the overlaid circle. These samples are then accumulated into orientation histograms summarizing the contents over 4x4 sub-regions, as shown on the right, with the length of each arrow corresponding to the sum of the gradient magnitudes near that direction within the region. The figure 5.4 shows a 2x2 descriptor array computed from an 8x8 set of samples, whereas the experiments in this paper use 4x4 descriptors computed from a 16x16 sample array.

A Gaussian weighting function with σ equal to one half the width of the descriptor window is used to assign a weight to the magnitude of each sample point. This is illustrated with a circular window on the left side of Figure 5.4, although, of course, the weight falls off smoothly. The

purpose of this Gaussian window is to avoid sudden changes in the descriptor with small changes in the position of the window, and to give less emphasis to gradients that are far from the center of the descriptor, as these are most affected by miss-registration errors. The keypoint descriptor is shown on the right side of Figure 5.4. It allows for significant shift in gradient positions by creating orientation histograms over 4x4 sample regions. The figure shows eight directions for each orientation histogram, with the length of each arrow corresponding to the magnitude of that histogram entry. A gradient sample on the left can shift up to 4 sample positions while still contributing to the same histogram on the right, thereby achieving the objective of allowing for larger local positional shifts. It is important to avoid all boundary affects in which the descriptor abruptly changes as a sample shifts smoothly from being within one histogram to another or from one orientation to another. Therefore, tri-linear interpolation is used to distribute the value of each gradient sample into adjacent histogram bins. In other words, each entry into a bin is multiplied by a weight of $1 - d$ for each dimension, where d is the distance of the sample from the central value of the bin as measured in units of the histogram bin spacing.

5.10 Keypoint matching

The best candidate match for each keypoint is found by identifying its nearest neighbor in the database of keypoints from training images. The nearest neighbor is defined as the keypoint with minimum Euclidean distance for the invariant descriptor vector as was described in Section 5.6. However, many features from an image will not have any correct match in the training database because they arise from background clutter or were not detected in the training images. Therefore, it would be useful to have a way to discard features that do not have any good match to the database. A global threshold on distance to the closest feature does not perform well, as some descriptors are much more discriminative than others. A more effective measure is obtained by comparing the distance of the closest neighbor to that of the second-closest neighbor. If there are multiple training images of the same object, then it can be defines as the second-closest neighbor as being the closest neighbor that is known to come from a different object than the first, such as by only using images known to contain different objects.

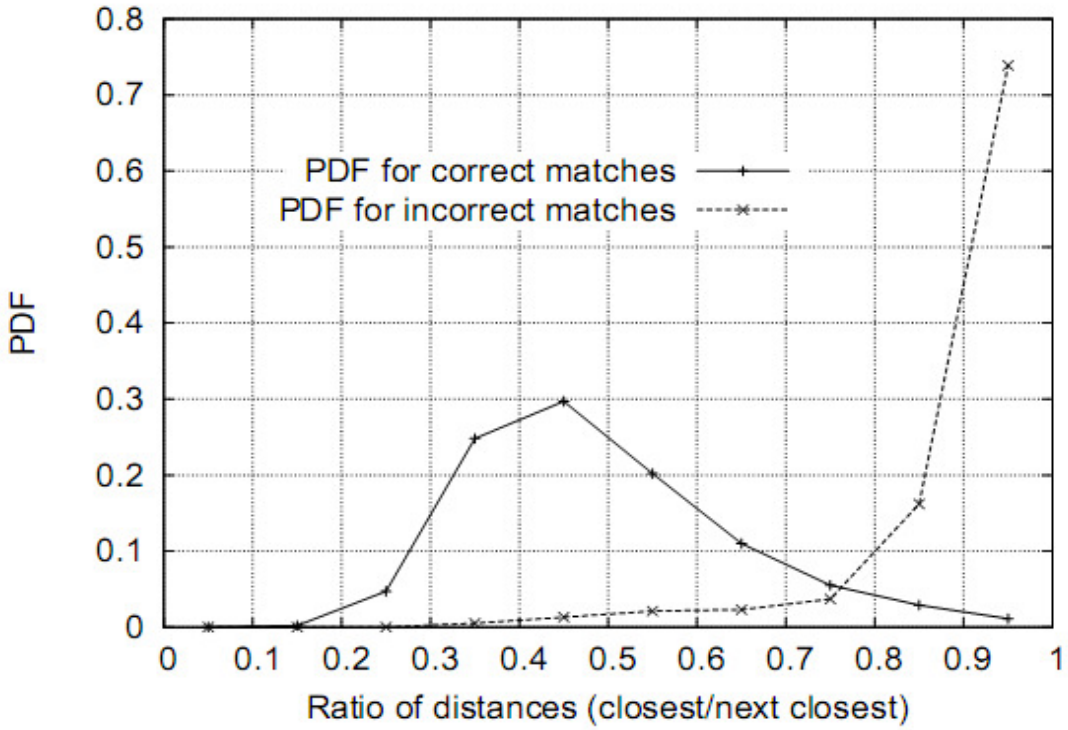


Figure 5.5 The probability that a match is correct can be determined by taking the ratio of distance from the closest neighbor to the distance of the second closest. Using a database of 40,000 keypoints, the solid line shows the PDF of this ratio for correct matches, while the dotted line is for matches that were incorrect.

Figure 5.5 shows the value of this measure for real image data. The probability density functions (PDF) for correct and incorrect matches are shown in terms of the ratio of closest to second-closest neighbors of each keypoint. Matches for which the nearest neighbor was a correct match have a PDF that is centered at a much lower ratio than that for incorrect matches. For our object recognition implementation, we reject all matches in which the distance ratio is greater than 0.8, which eliminates 90% of the false matches while discarding less than 5% of the correct matches. This figure was generated by matching images following random scale and orientation change, a depth rotation of 30 degrees, and addition of 2% image noise, against a database of 40,000 keypoints.

CHAPTER 6

PROPOSED WORK

6.1 Proposed Approach

This chapter converse the proposed approach and pores matching using SIFT algorithm. Figure 6.1 shows the block diagram of proposed approach. The basics of SIFT technique is described in the previous chapter. In this section, the proposed algorithm, its features and other various aspects has been described. Two type of database has been created; first one name samedb1 contains the 10 fingerprints of same person with some variations and other factors such as light, noise etc. Thus samedb1 contains 400 fingerprints of 40 different students. Second database namely diffdb2 contains 150 fingerprints of different students.

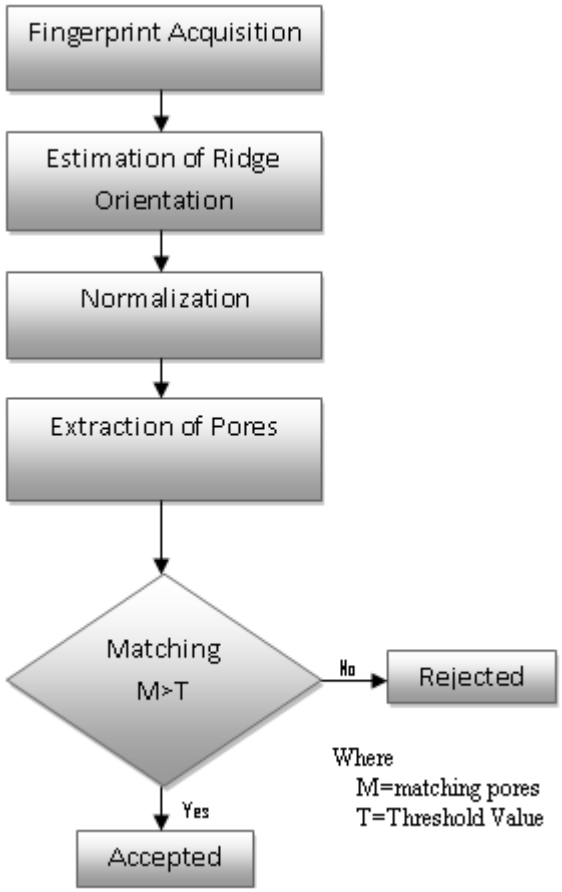


Figure 6.1 Block diagram of proposed approach

6.2 Fingerprint Acquisition and Database

The first step in the proposed approach is to acquire fingerprint image of good quality. Thus, Hamster II is use for acquiring fingerprint image. Hamster II is optical fingerprint scanner and use for scanning the finger. Hamster II is used for creating fingerprint database. This database is use for analyzing the accuracy of proposed algorithm and execute the results on the basis of analyze. After acquiring fingerprint, store it in database and also use as input to proposed algorithm. Figure 6.2 shows the acquired fingerprint from optical scanner.

Figure 6.2 Acquired fingerprint from optical scanner

6.3 Estimation of ridge orientation

Next process is the estimation of the ridge orientation. The local ridge orientation is determined by the least square estimate method. This data is utilized later in the representation of pores. Analysis of the developed fingerprint matching system has revealed a number of interesting conclusions. It can be stated that segmentation is the critical stage of fingerprint pores recognition, since areas that are wrongly identified as pores regions will corrupt biometric

templates resulting in very poor recognition. Segmentation can be the most difficult stage of pores recognition because its success is dependent on the imaging quality of fingerprint images. 95% of the fingerprint database images segmented correctly. Another interesting finding was that the encoding process only required one Gabor filter to provide accurate recognition, since the open literature mentions the use of multi-scale representation in the encoding process.

Figure 6.3 Ridge orientation of fingerprint

6.4 Normalization

To compensate for the variations in lighting, contrast and other inconsistencies, three preprocessing steps are used: Gaussian blur, sliding window contrast adjustment, and histogram based intensity level correction. Gaussian blurring is used to remove any noise introduced by the sensor. Normalizes image values to 0-1, or to desired mean and variance. Offsets and rescales image so that the minimum value is 0 and the maximum value is 1. Result is returned in n. If the image is color the image is converted to HSV and the value/intensity component is normalized to 0-1 before being converted back to RGB.

The lighting inconsistencies are adjusted by using sliding-window contrast adjustment on the

Gaussian blurred image. To further enhance the ridges and valley a final intensity correction is made by using Histogram-based Intensity Level Adjustment.

The image can divide into small processing blocks (32 by 32 pixels) and perform the Fourier transform according to:

$$F(u, v)=\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x, y) \times \exp\left\{-j2\pi\times\left(\frac{ux}{M}+\frac{vy}{N}\right)\right\}$$

…… (6.1)

For u = 0, 1, 2... 31 and v = 0, 1, 2... 31.

In order to enhance a specific block by its dominant frequencies, multiply the FFT of the block by its magnitude a set of times. Where the magnitude of the original FFT = abs (F(u, v)) = |F(u, v)|.

Get the enhanced block according to

$$g(x, y) = F^{-1}\{F(u, v)\times|F(u, v)|^{k}\}$$

…… (6.2)

Where $F^{-1}(F(u, v))$ is done by:

$$F(x, y)=\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x, y) \times \exp\left\{j2\pi\times\left(\frac{ux}{M}+\frac{vy}{N}\right)\right\}$$

……. (6.3)

For x = 0, 1, 2... 31 and y = 0, 1, 2... 31.

The k in formula (6.2) is an experimentally determined constant, which can choose k=0.45 to calculate. While having a higher "k" improves the appearance of the ridges, filling up small holes in ridges, having too high a "k" can result in false joining of ridges. Thus a termination might become a bifurcation.

The enhanced image after FFT has the improvements to connect some falsely broken points on ridges and to remove some spurious connections between ridges. The side effect of each block is obvious but it has no harm to the further operations because resultant image after consecutive binarization operation is pretty good as long as the side effect is not too rigorous

For the fingerprint image preprocessing stage, Fourier Transform can be use to image enhancement. And then the fingerprint image is binarized using the locally adaptive threshold method. The image segmentation task is fulfilled by a three-step approach: block direction

estimation, segmentation by direction intensity and Region of Interest extraction by Morphological operations. Most methods used in the preprocessing stage are developed by other researchers but they form a brand new combination in our project through trial and error. Also the morphological operations for extraction ROI are introduced to fingerprint image segmentation in this thesis.

Figure 6.4 Normalized image

6.5 Pores estimation and extraction approach

Extract level 3 features in ROI. The pores are distributed over ridges and using orientation detail can provide additional information for matching. During tracing, the algorithm classifies the contour information into pores and ridges.

A blob of size greater than 2 pixels and less than 45 pixels is classified as a pore. Therefore,

noisy contours, which are sometimes wrongly extracted, are not included in the feature set. A pore is approximated with a circle and the center is used as the pore feature. The fingerprint image is threshold with a single-point threshold (T). After this step the pores have an intensity of 255. The pores are then extracted by a blob detector which locates groups of connected pixels (pores) with an intensity of 255 and with size within a pre-determined range. Each pore thus extracted is represented by the coordinates of the central pixel and an orientation, which is the ridge orientation at that particular location.

An edge of a ridge is defined as the ridge contour. Each row of the ridge feature represents x; y coordinates of the pixel and direction of the contour at that pixel. Here edges will remove by using morphological operation and extract only pores whose pixel value is greater than 15. In other words, there is assumption that pores can be classified as combination of 2 or more pixels. Using this assumption, we consider only those pores who are grouping of more than 15 but less than 45 pixels. Rest pixels will remove from normalized image. Thus it is possible to remove the contour ridges and extract pores. Figure 6.4 shows the extracted pores from normalized image.

The pixel intensity values in the fingerprint image are typically non-invariant over the time of capture and there is need to determine salient feature of input fingerprint image that can be discriminate between identities as well as remain invariant for a given individual. Thus the problem of representation is to determine a measurement (features) space in which fingerprint image belonging to the same finger form a compact cluster and those belonging to the different finger occupy different portions of space.

The last step is to remove possible spurious pores. We apply the following constraints to post-processing the initial pore extraction results.

(I) Pores should reside on ridges only. To implement this constraint, we use the binary ridge image as a mask to filter the extracted pores.

(II) Pores should be within a range of valid sizes. We measure the size of a pore by counting the pixels in its region.

(III) The mean intensity of a true pore should be large enough. In our experiments, we discarded the last 5% pores (i.e. those with lowest intensity). Finally, we record the extracted pores' locations as the coordinates of their mass centers.

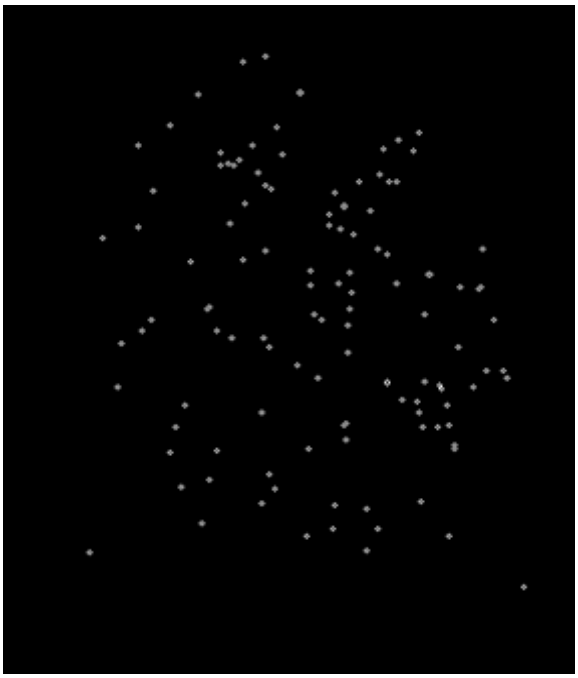


Figure 6.4 Extracted pores from normalized image

6.6 Pores based fingerprint matching

In latent print comparison, when Level 1 or Level 2 features are similar between the template and the query, a forensic expert often investigates Level 3 details. To be compatible with current AFIS systems, our matching using Level 2 and Level 3 features is done separately, except using the Level 2 information (minutiae) for initial alignment for Level 3 matching. Then a score-level fusion of both the matching stages is performed using the sum rule and min-max normalization. The best candidate match for each keypoint is found by identifying its nearest neighbor in the database of keypoints from training images. The nearest neighbor is defined as the keypoint with minimum Euclidean distance for the invariant descriptor vector as was described in Chapter 5. A global threshold on distance to the closest feature does not perform well, as some descriptors are much more discriminative than others.

A more effective measure is obtained by comparing the distance of the closest neighbor to that of the second-closest neighbor. If there are multiple training images of the same object, then it can be defined the second-closest neighbor as being the closest neighbor that is known to come from a different object than the first, such as by only using images known to contain different objects. This measure performs well because correct matches need to have the closest neighbor

significantly closer than the closest incorrect match to achieve reliable matching. For false matches, there will likely be a number of other false matches within similar distances due to the high dimensionality of the feature space. We can think of the second-closest match as providing an estimate of the density of false matches within this portion of the feature space and at the same time identifying specific instances of feature ambiguity. No algorithms are known that can identify the exact nearest neighbors of points in high dimensional spaces that are any more efficient than exhaustive search.

Our keypoint descriptor has a 128-dimensional feature vector, and the best algorithms, such as the k-d tree provide no speedup over exhaustive search for more than about 10 dimensional spaces. Therefore, we have used an approximate algorithm, called the Best-Bin-First (BBF) algorithm Lowe [41]. This is approximate in the sense that it returns the closest 20 neighbor with high probability. The BBF algorithm uses a modified search ordering for the k-d tree algorithm so that bins in feature space are searched in the order of their closest distance from the query location. In this thesis, we cut off search after checking the first 200 nearest-neighbor candidates. One reason the BBF algorithm works particularly well for this problem is that we only consider matches in which the nearest neighbor is less than 0.8 times the distance to the second-nearest neighbor (as described in the previous section), and therefore there is no need to exactly solve the most difficult cases in which many neighbors are at very similar distances.

Figure 6.5 shows the pores matching result. As we set threshold value i.e. 25, when SIFT matching more than 25 keypoints then it can be accepted fingerprint image otherwise fingerprint will be rejected.

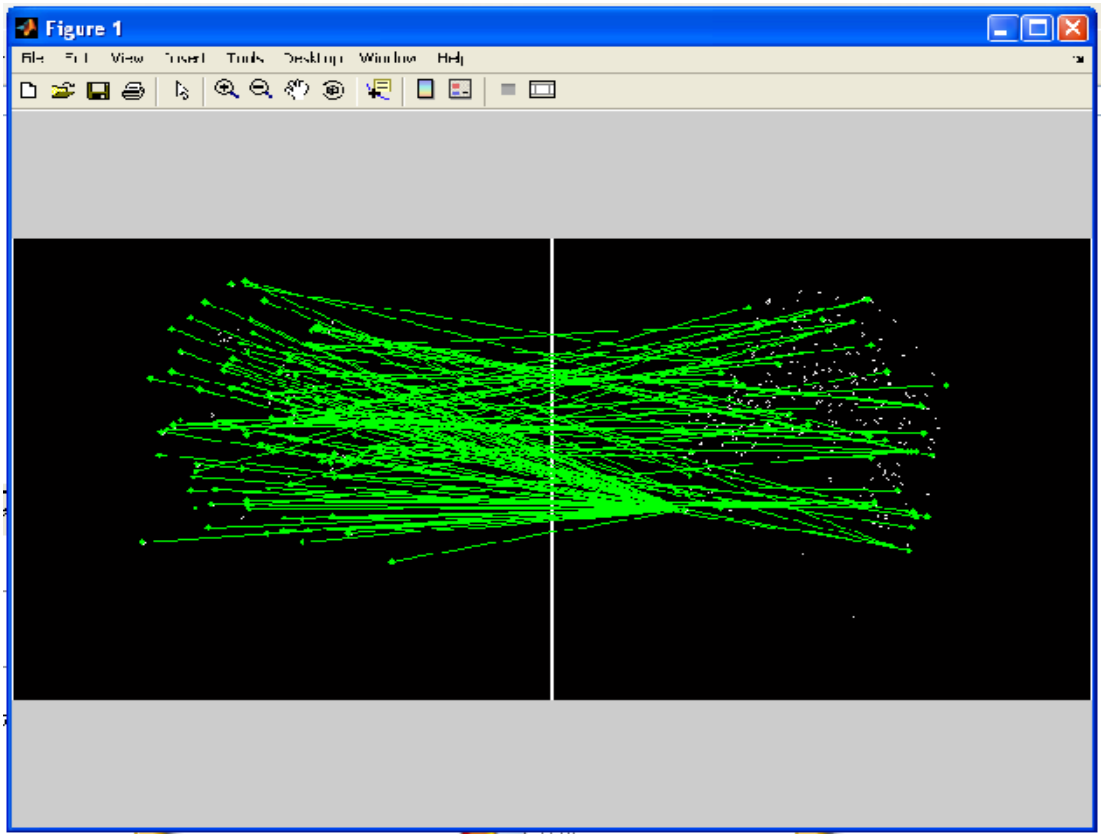
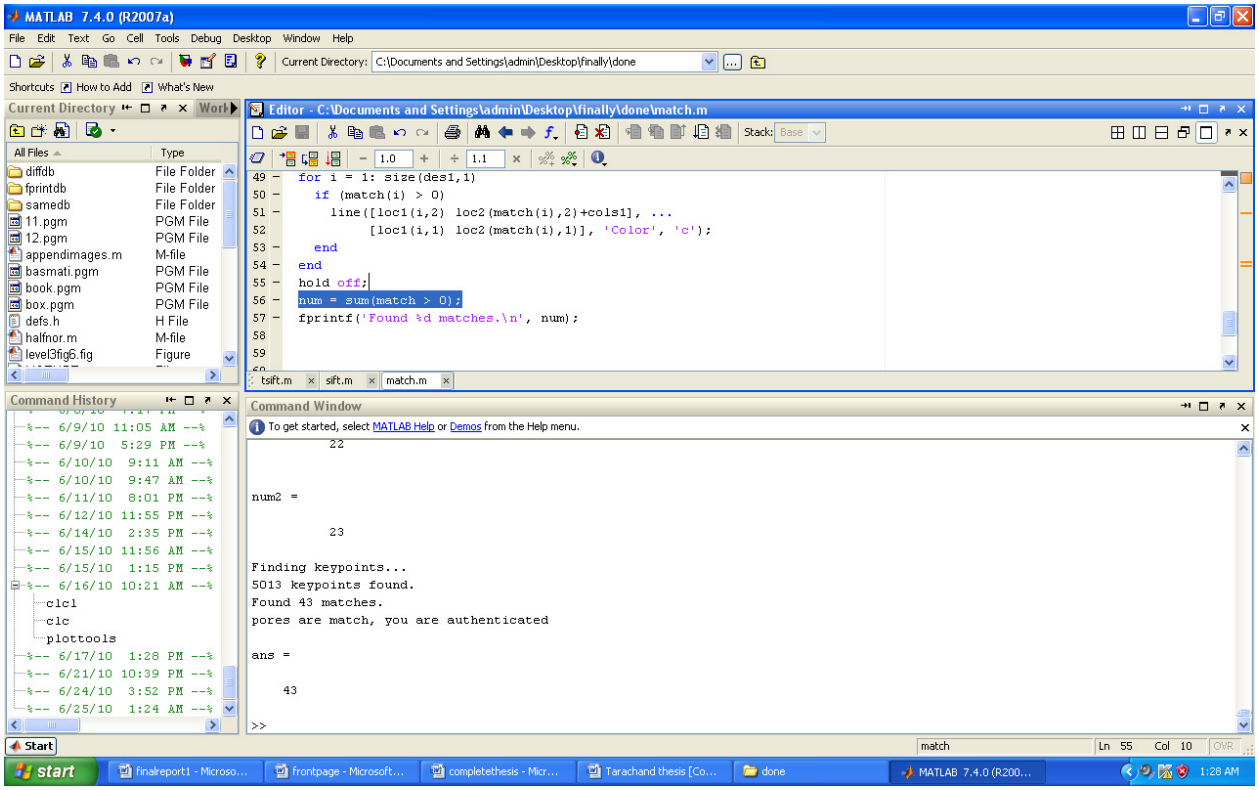


Figure 6.5 Matching Pores (keypoints) and Output (below)

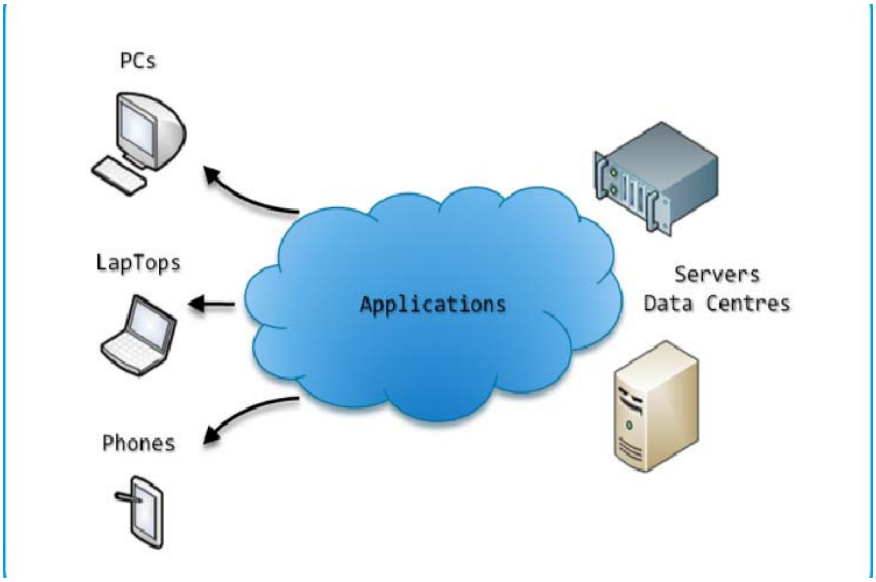


6.7 Cloud Computing Introduction

Cloud Computing has been envisioned as the next-generation architecture of IT enterprise, due to its long list of unprecedented advantages in the IT history: on-demand self-service, ubiquitous network access, location independent resource pooling, rapid resource elasticity, usage-based pricing and transference of risk [50]. As a disruptive technology with profound implications, Cloud Computing is transforming the very nature of how businesses use information technology. From users' perspective, including both individuals and IT enterprises, storing data remotely into the cloud in a flexible on-demand manner brings appealing benefits relief of the burden for storage management, universal data access with independent geographical locations, and avoidance of capital expenditure on hardware, software, and personnel maintenance, etc [51].
Cloud computing is a recent trend in IT that moves computing and data away from desktop and portable PCs into large data centers. It refers to applications delivered as services over the Internet as well as to the actual cloud infrastructure — namely, the hardware and systems software in data centers that provide these services. The key driving forces behind cloud computing are the ubiquity of broad- band and wireless networking, falling storage costs, and progressive improvements in Internet computing software. Cloud-service clients will be able to add more capacity at peak demand, reduce costs, experiment with new services, and remove unneeded capacity, whereas service providers will increase utilization via multiplexing, and allow for larger investments in software and hardware.
Currently, the main technical under pinning's of cloud computing infrastructures and services include virtualization, service-oriented software, grid computing technologies, management of large facilities, and power efficiency. Consumers purchase such services in the form of infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS), or software-as-a-service (SaaS) and sell value added services (such as utility services) to users. Within the cloud, the laws of probability give service providers great leverage through statistical multiplexing of varying workloads and easier management a single software installation can cover many users' needs.

We can distinguish two different architectural models for clouds: The first one is designed to scale out by providing additional computing instances on demand. Clouds can use these instances to supply services in the form of SaaS and PaaS. The second architectural model is designed to provide data and compute-intensive applications via scaling capacity. In most cases, clouds provide on-demand computing instances or capacities with a "pay-as-you-go" economic model. The cloud infrastructure can support any computing model compatible with loosely coupled CPU clusters. Organizations can provide hardware for clouds internally (internal clouds), or a third party can provide it externally (hosted clouds). A cloud might be restricted to a single organization or group (private clouds), available to the general public over the Internet (public clouds), or shared by multiple groups or organizations (hybrid clouds). A cloud comprises processing, network, and storage elements, and cloud architecture consists of three abstract layers. Infrastructure is the lowest layer and is a means of delivering basic storage and

compute capabilities as standardized services over the network. Servers, storage systems, switches, routers, and other systems handle specific types of workloads, from batch processing to server or storage augmentation during peak loads. The middle platform layer provides higher abstractions and services to develop, test, deploy, host, and maintain applications in the same integrated development environment. The application layer is the highest layer and features a complete application offered as a service.



Cloud Computing delivers (amongst others) the following advantages:

1. The cost to the end user may be reduced as Cloud Computing vendors operate the entire architecture and users need only have access to the internet via a suitable client side application such as a web browser.
2. All data and processes are server side so Cloud Applications can be accessed from any computer connected to the internet whether it be a private PC, a public PC or mobile PDA/Smart Device.
3. Cloud Applications can be hosted alongside each other in centralized data centers. By sharing resources such as processing, memory and bandwidth, costs may be reduced while performance, efficiency and scalability may be increased.
4. The security of end users data is typically higher due to the centralization of data within data centers that have hive like security concerns beyond those of any individual Cloud Application.

CHAPTER 7

RESULTS

In this Chapter, we have analyzed the Performance of the proposed technique known as pores matching using SIFT algorithm. As this is a novel approach, so for this reason we have chosen the images of fingerprint from Hamster fingerprint scanner. We have taken here total 100 images, and consider two to eight images of each finger with respect to variations in images for analysis and key points detection same user, so it will become 500 fingerprint images are collected in a database.

We create two types of database; first one name samedb1 contains the 10 fingerprints of same person with some variations and other factors such as light, noise etc. Thus samedb1 contains 400 fingerprints of 40 different students. Second database namely diffdb2 contains 150 fingerprints of different students. These fingerprints acquire using Hamster II optical fingerprint scanner. Fingerprints are acquired after taking some interval of time. There is another database name fprintdb that was downloaded from FVC 2002 for analysis.

Experimental results are obtained using the cross validation approach. We perform experiments by evaluation of the proposed level-3 feature extraction algorithm. We first compute the verification performance of the proposed level-3 feature extraction algorithm and compare it with existing level-3 feature based verification algorithms.

7.1 Genuine Acceptance Rate

The graph plots in Figure 7.2 and Genuine Acceptance Rate in Table 7.1 summarize the results of this experiment and comparison the results with other existing approaches given by other researchers. The proposed level-3 feature extraction algorithm yields a verification accuracy of 93.41% which is 2–7% better than existing algorithms.

Table 7.1 GAR of proposed approach

Threshold	25	35	45	70
Accepted images	188	181	149	82

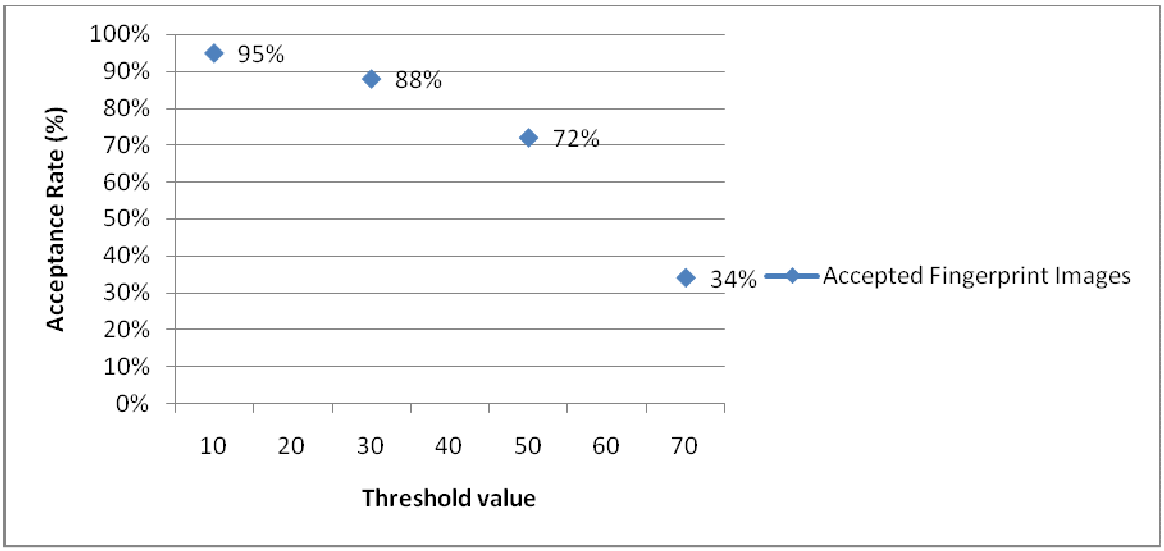


Figure 7.1 GAR graph of proposed approach

Finally we will take another database in account to compare the proposed results with existing results, for this we have chosen database of images from FVC 2002 and examining the proposed technique with it. So the result shows that the proposed technique will give better results.

Table 7.2 Comparison of Genuine Acceptance Rate (GAR) with other existing approaches.

(threshold value: 40)

jain's	Kryszczuk's	Level 2	M. Vatsa	Proposed
88.07	84.03	89.11	91.41	93

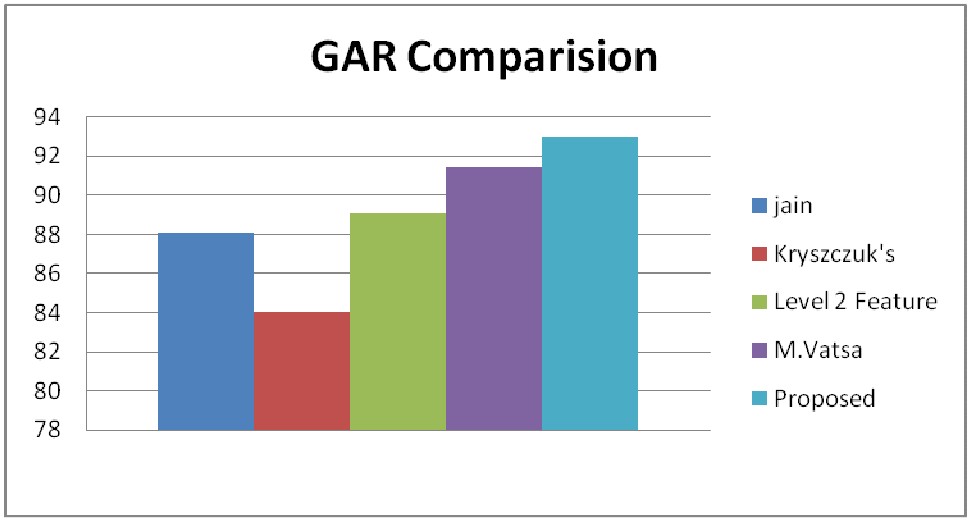


Figure 7.2 Comparison Graph

7.2 False Rejection Rate:

Fraction of attempts for which a fingerprint system denies access to a valid user. Each sample in a database is matched against the remaining samples of the same finger to compute the False Rejection Rate (FRR). To calculate FRR, we use samedb1 database. The FRR is the fraction of genuine fingerprints which are rejected and is calculated as follows

$$\text{FRR} = \frac{\text{Number of genuine fingerprints rejected}}{\text{Total number of genuine tests}}$$

The total number of genuine tests is 115 and 215 for Hamster II database, FVC 2002 Database respectively. Table 3 summarizes the FRR and Figure 7.3 shows the graph of FRR of proposed approach. As table 3 summarize the result of this analyzing process, we conclude that, as the threshold value is increases, false rejection rate is also increases.

Table 7.3 False Rejection Rate (FRR) of proposed approach.

Threshold	25		35		45		70
FRR	7%		12%		28%		66%

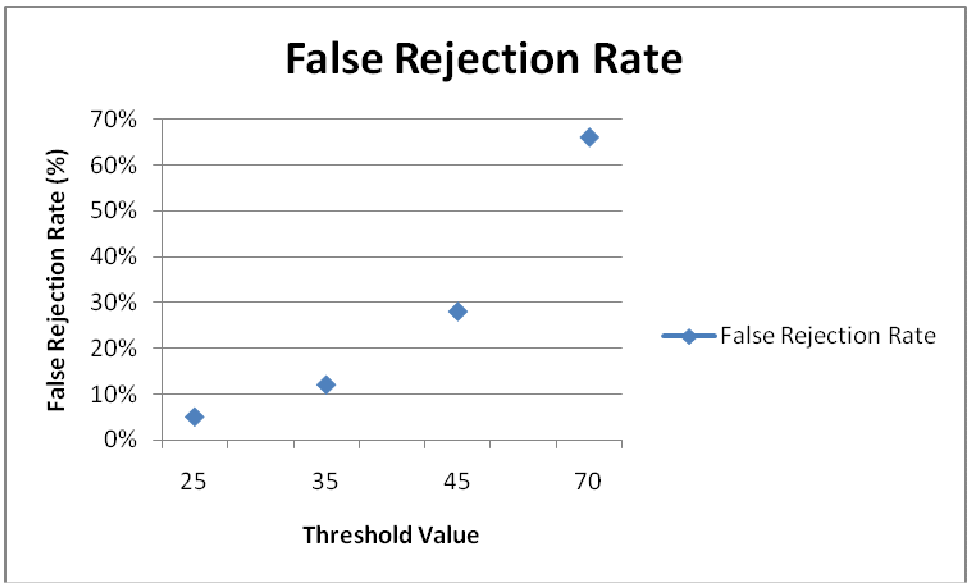


Figure 7.3 False Rejection Rate of proposed approach.

7.3 False Acceptance Rate

For all databases the first sample of each finger is matched against the first sample of the remaining different fingers in the database to compute the False Acceptance Rate (FAR). To calculate FAR, we use diffdb2 database. The FAR is the fraction of impostor fingerprints which are accepted and is calculated as follows

$$\text{FAR} = \frac{\text{Number of impostor fingerprints accepted}}{\text{Total number of impostor tests}}$$

If the matching g is performed, the symmetric one (i.e., h against g) is not executed to avoid correlation. FAR of proposed approach is analysis in table 4 and graph is shown in Figure 7.4

As table 4 summarize the result of this analyzing process, we conclude that, as the threshold value is increases, false acceptance rate is decreases.

Table 7.4 FAR of proposed approach.

Threshold	FAR (%)
25	58
35	36
45	15
55	5

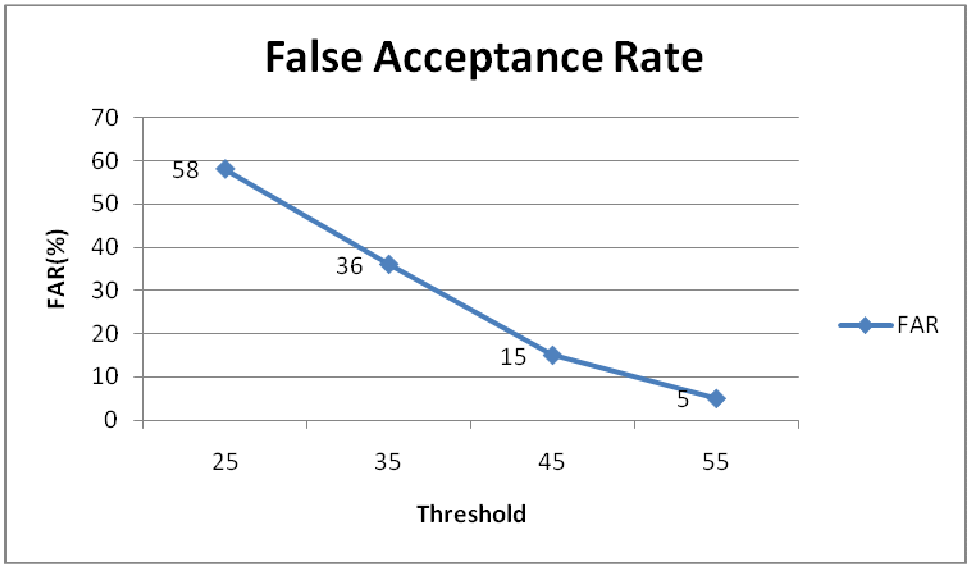


Figure 7.4 False Acceptance Rate of proposed approach

CHAPTER 8

CONCLUSION AND FUTURE WORK

8.1 CONCLUSION

This dissertation presents a concise introduction of fingerprint level 3 features extraction and matching approach which is a novel approach, its characteristics, design issues and applications. It also describes an overview of Level 1 and level 2 features, in the literature and their functionalities. Along with that, it has through discussion of SIFT algorithm. Finally, it presents an novel approach for level 3 feature extraction and matching algorithm. Since the technology is going to its zenith, the way of its journey is not smooth and many loopholes are there.

The proposed work is an attempt to overcome some weakness regarding security concern of the system. The experimental results demonstrate that the proposed approach and its associated pore extraction method can detect pores more accurately and robustly, and can help to improve the verification accuracy of pore based fingerprint recognition systems. There are many method exists to make it unextractable by adversaries, like one is to use multi-biometric traits under a single process, but it need extra sensors setup for each kind of traits, the proposed technique gives an extra edge to such systems which use single type of sensor and give more security.

This approach will give the technology new amplitude in order to provide a secure way of authentication, in which the pores are logically extracted. The proposed algorithm performs better than existing recognition algorithms and fusion algorithms. Along with other advantages, in all biometric systems fingerprint based systems are more efficient than other multimodal system, so it minimizes FAR and FRR.

8.2 FUTURE WORK

The Proposed technique can give an extra edge to biometric systems credibility; it will reduce the security overhead and burden because a level 3 feature provides more precise information that use for authentication purpose. Future work will also characterize the performance of level-3 features on a comprehensive large scale database which contains fingerprint images of varying size, quality and other environmental factors. Since the level 3 features are unique so it

may give more security with minimal error to defense, corporate & other major organizations, where security is the main concern. Beside this to speedup the matching process cloud computing concept will be introduced. Cloud computing has evolved as an emerging technology in the field of computer science. Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to adjust to a variable load (scale), allowing also for an optimum resource utilization. The voter's fingerprint database is huge in size. To check the authentication of voter, input fingerprint template has to be matched against entire stored fingerprint database. To increase the speedup and efficiency gain fingerprint database will be uploaded on cloud where matching results can be retrieved fastly as compare to matching on a single high computing power CPU. On cloud computing entire database is divided into number of blocks which runs on different nodes independently and significantly gain can be achieved.

References:

1. Schneier, Bruce. Applied Cryptography. New York: John Wiley & Sons, 1996.
2. M. Ray, P. Meenen, and R. Adhami, "A novel approach to fingerprint pore, extraction." Southeastern Symposium on System Theory, page no. 282–286, 2005.
3. McGraw, Gary and Greg Morrisett., "Attacking Malicious Code: A Report to the Infosec Research Council." IEEE Software. September/October 2000.
4. The Implementation of Electronic Voting in the UK research summary. 2002. "http://www.dca.gov.uk/elections/e-voting/pdf/e-summary.pdf." 21.01.2007.
5. D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, and A.K. Jain. FVC2000: Fingerprint Verification Competition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(3):402–412, 2002.
6. Qijun Zhao, Lei Zhang, David Zhang, Nan Luo, "Adaptive Pore Model for Fingerprint Pore Extraction." Proc. IEEE, 978-1-4244-2175-6/08, 2008.
7. Moheb R. Girgis, Tarek M. Mahmoud, and Tarek Abd-El-Hafeez, "An Approach to Image Extraction and Accurate Skin Detection from Web Pages." World academy of Science, Engineering and Technology, page no. 27, 2007.
8. Manvjeet Kaur, Mukhwinder Singh, Akshay Girdhar, and Parvinder S. Sandhu, "Fingerprint Verification System using Minutiae Extraction Technique." World academy of Science, Engineering and Technology, page no. 46, 2008.
9. Hoi Le, The Duy Bui, "Online fingerprint identification with a fast and distortion tolerant hashing." Journal of Information Assurance and Security 4 page no. 117-123, 2009.
10. Anil Jain, Yi Chen, and Meltem Demirkus, "Pores and Ridges: Fingerprint Matching Using Level 3 Features." Pattern recognition letters, page no. 2221-2224, 2004.
11. Mayank Vatsa, Richa Singh, Afzel Noore, Sanjay K. Singh, "Combining pores and ridges with minutiae for improved fingerprint verification." Elsevier, Signal Processing 89, page no. 2676–2685, 2009.
12. Qijun Zhao, Lei Zhang, David Zhang, Nan Luo, "Adaptive Pore Model for Fingerprint Pore Extraction." IEEE, 978-1-4244-2175, 2008.
13. A.K. Jain, R. Bolle, S. Pankanti (Eds.), "Biometrics: Personal Identification in Networked Society", Kluwer Academic Publishers, Dordrecht, 1999.
14. Umut Uludaga, Arun Rossb, Anil Jain, "Biometric template selection and update: a case study in fingerprints." U. Uludag et al. / Pattern Recognition 'Elsavier', 37 page no. 1533 – 1542, 2004.
15. Anil K. Jain and David Maltoni., "Handbook of Fingerprint Recognition." Springer Verlag New York, Inc., Secaucus, NJ, USA, 2003.
16. K. Kryszczuk, P. Morier, and A. Dryga jlo., "Study of the Distinctiveness of Level 2 and Level 3 Features in Fragmentary Fingerprint Comparison." In Proc. Of Biometric Authentication Workshop, page no. 124–133, May 2004.
17. K. Kryszczuk, A. Drygajlo, and P. Morier, "Extraction of Level 2 and Level 3 features for

fragmentary fingerprints." Proc. of the 2nd COST275 Workshop, Vigo, Spain, page no. 83-88, 2004.
18. D. Maio, D. Maltoni, A. K. Jain, and S. Prabhakar., "Handbook of Fingerprint Recognition." Springer Verlag, 2003.
19. http://www.itl.nist.gov/iad/894.03/fing/summary.html, NIST Fingerprint Data Exchange Workshop, 1998.
20. S. Pankanti, S. Prabhakar, and A. K. Jain, "On the Individuality of Fingerprints." IEEE Trans. PAMI, Vol. 24, page no. 1010-1025, 2002.
21. J.D. Stosz and L.A. Alyea, "Automated system for fingerprint authentication using pores and ridge structure." Proc. of the SPIE Automatic Systems for the Identification and Inspection of Humans, Volume 2277, page no. 210-223, 1994.
22. P.J. Besl and N.D. McKay, "A method for registration of 3-D shapes." IEEE Trans. PAMI, Vol. 14, page no. 239-256, 1992.
23. A. Tsai, A. Yezzi Jr., A. Willsky, "Curve evolution implementation of the Mumford–Shah functional for image segmentation, de-noising, interpolation, and magnification." IEEE Transactions on Image Processing 10 (8) page no. 1169–1186, 2001.
24. T. Chan, L. Vese, "Active contours without edges." IEEE Transactions on Image Processing, 10 (2) page no. 266–277, 2001.
25. A.R. Roddy and J.D. Stosz, "Fingerprint features–statistical analysis and system performance estimates" Proc. IEEE, vol. 85, no. 9, page no. 1390-1421, 1997.
26. http://www.fmrib.ox.ac.uk/~steve/susan/thinning/node2.html
27. Q. Zhang and K. Huang, "Fingerprint classification based on extraction and analysis of singularities and pseudo ridges." 2002.
28. http://www.owlnet.rice.edu/~elec301/Projects00/roshankg/elec301.htm
29. A. Luk, S.H. Leung, "A Two Level Classifier For Fingerprint Recognition." in Proc. IEEE 1991 International Symposium on CAS, Singapore, page no. 2625-2628, 1991.
30. N. K. Ratha, K. Karu, S. Chen, and A. K. Jain, "A real-time matching system for large fingerprint databases." Transactions on Pattern Analysis and Machine Intelligence, 18(8): page no. 799–813, 1996.
31. L. Coetzee and E. C. Botha, "Fingerprint recognition in low quality images." Pattern Recognition, 26(10), 1993.
32. D. Marr and E. C. Hilderith, "Theory of edge detection." Proceedings of the Royal Society, pages 187–217, 2004.
33. L. O'Gormann and J.V.Nickerson, "An approach to fingerprint filter design." Pattern Recognition, 22(1): page no. 29–38, 1989.
34. Greenberg S., Aladjem M., Kogan D., and Dimitrov I. "Fingerprint image enhancement using filtering techniques." In International Conference on Pattern Recognition, volume 3, page no 326–329, 2000.
35. Ruud M. Bolle, Sharath Pankanti and Nalini K. Ratha, "Evaluation techniques for biometrics-based authentication systems (FRR)." IBM Thomas J. Watson Research Center.

36. Wang Yuan, Yao Lixiu, Zhou Fuqiang, "A real time fingerprint recognition system based on novel fingerprint matching strategy." The eighth international conference on electronic measurement and instruments, ICEMI 2007.
37. Wei Cui, Guoliang Wu, Rongjin Hua, and Hao Yang, "The Research of Edge Detection Algorithm for Fingerprint Images." IEEE' 2008.
38. Shunshan li, Min Wei, Haiying Tang, Tiange Zhuang and Michael H. Buonocore, "Image Enhancement Method for Fingerprint Recognition System.", Proceedings of the 2005 IEEE, Engineering in Medicine and Biology 27th Annual Conference, Shanghai, China, September 1-4, page no. 3386-3389, 2005.
39. S. Mil'shtein, A. Pillai, A. Shendye, C. Liessner, and M. Baier, "Fingerprint Recognition Algorithms for Partial and Full Fingerprints." IEEE 2008.
40. Arun Ross, Umut Uludag, Anil Jain, "Biometric template selection and update: a case study in Fingerprints." Pattern Recognition Society. Published by Elsevier Ltd., 2003.
41. Deepak Kumar Karna, Suneeta Agarwal, Shankar Nikam, "Normalized Cross-correlation based Fingerprint Matching." Fifth International Conference on Computer Graphics, Imaging and Visualization, IEEE 2008.
42. Asker M. Bazen, Gerben T.B. Verwaaijen, Sabih H. Gerez, "A Correlation-Based Fingerprint Verification System." Workshop on Circuits, Systems and Signal Processing, Veldhoven, The Netherlands, November 2000.
43. David G. Lowe, "Distinctive Image Features from Scale-Invariant Key points." International Journal of Computer Vision, 2004.
44. Working draft of CDEFFS: the ANSI/NIST committee to define an extended fingerprint feature set, 2008. Available at "http://fingerprint.nist.gov/standard/cdeffs/index.html".
45. Lin Zhang, Lei Zhang, David Zhang, Hailong Zhu, "Online finger-knuckle-print verification for personal authentication." Pattern recognition, Elsevier, 2010.
46. D. Lowe "Distinctive Image Features from scale-invariant keypoints" *Int. J. Comp. Vision*, 2004.
47. X.D. Jiang, W.Y. Yau, W. Ser, "Detecting the fingerprint minutiae by adaptive tracing the gray level ridge, Pattern Recognition." page no. 999–1013, 2001.
48. A. Jain, R. Bolle, L. Hong, "Online fingerprint verification." IEEE Transactions on Pattern Analysis and Machine Intelligence, page no. 302–314, 1997.
49. FVC2002. http://bias.csr.unibo.it/fvc2002/.
50. 50.]P. Mell and T. Grance, "Draft nist working definition of cloud computing," Referenced on June. 3rd, 2009 Online at http://csrc.nist.gov/ groups/SNS/cloud-computing/index.html, 2009.
51. 51.M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Kon winski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia,"Above the clouds: A berkeley view of cloud computing," University of California, Berkeley, Tech. Rep. UCB-EECS-2009-28, Feb 2009.