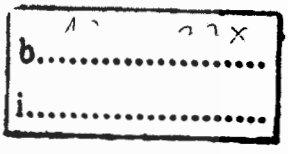
FINGERPRINT IMAGES ENHANCEMENT WITH DIRECTIONAL FILTERING

KEOKANLAYA SIHALATH



A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF ENGINEERING IN ELECTRONICS ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2010

KMITL-2010-EN-M-040-038

หัวข้อวิทยานิพนธ์	การปรับปรุงภาพลายพิมพ์นิ้วมือโดยใช้ตัวกรองแบบมีทิศทาง
นักศึกษา	นายแก้วกัลยา สีหราช
รหัสนักศึกษา	51060418
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมอิเล็กทรอนิกส์
พ. ศ.	2553
อาจารย์ที่ปรึกษาวิทยานิพนธ์	รศ. ดร. สมศักดิ์ ชุมช่วย

บทคัดย่อ

ความก้าวหน้าในการพิสูจน์ตัวบุคคลด้วยคุณลักษณะทางชีวภาพได้มีคืบหน้าอย่างมาก เพราะต้องการความน่าเชื่อถือและทำให้ง่ายต่อระบบการรักษาความปลอดภัยและประหยัดต้นทุน ลักษณะทางชีวภาพซึ่งมีมากมายนั้น ลายพิมพ์นิ้วมือก็เป็นหนึ่งในนั้นที่มีความน่าเชื่อถือสูงในการพิสูจน์ตัวบุคคล เพราะแต่ละตัวบุคคลจะมีลักษณะลายพิมพ์นิ้วมือที่แตกต่างกันไป ไม่เพียงแต่ในกระบวนการพิสูจน์ตัวบุคคลเท่านั้น ลายพิมพ์นิ้วมือได้ถูกนำมาใช้อย่างกว้างขวางในกระบวนการยืนยันตัวบุคคลด้วย ดังนั้นวัตถุประสงค์สูงสุดก็คือ การจับคู่เหมือนลายพิมพ์นิ้วมือนั่นเอง อย่างไรก็ตามเพื่อให้ได้มาซึ่งความรวดเร็วและมีความน่าเชื่อถือสูงในขั้นตอนการตรวจสอบนั้น ทั้งการปรับปรุงลายพิมพ์นิ้วมือและการคัดแยกลายพิมพ์นิ้วมือจะเป็นขั้นตอนที่ไม่อาจหลีกเลี่ยงได้ ซึ่งบ่อยครั้งภาพลายพิมพ์นิ้วมือที่ได้มานั้นยังขาดความครบถ้วนชัดเจน และหากภาพที่ได้มานั้นมีความไม่สมบรูณ์มากหรือมีคุณภาพต่ำก็จะทำให้ทำให้การจดจำ หรือการแยกแยกที่ต่ำไปด้วย โดยภาพที่ได้มานั้นจะสมบูรณ์หรือไม่สมบูรณ์ ในบางครั้งก็หลีกเลี่ยงไม่ได้ เช่นอาจจะเลือน หรือไม่ก็เลอะจนเกินไป และในบางครั้งก็เป็นลายพิมพ์นิ้วมือที่ควบคุมคุณภาพการนำเข้าไม่ได้ เช่น ลายพิมพ์นิ้วมือของคนร้ายที่เก็บจากที่เกิดเหตุ ซึ่งปกติแล้วก็เก็บได้เพียงบางส่วนและมีคุณภาพต่ำอีกด้วย ในสภาวะแวดล้อม และเงื่อนไขต่างๆ เหล่านี้ การปรับปรุงภาพลายพิมพ์นิ้วมือจึงมีความสำคัญ และมากพอที่ทำให้เรามีความสนใจที่จะทำการศึกษา

ในวิทยานิพนธ์ฉบับนี้ผู้เขียนได้ทำการศึกษาวิธีการปรับปรุงคุณภาพของภาพลายพิมพ์นิ้วมือในขอบเขตของบางวิธีเท่านั้น โดยจะเน้นไปยังตัวกรองแบบมีทิศทางสองตัวตัวหลัก คือตัวกรองชนิดกการ์บอร์ และตัวกรองอนุพันธ์อันดับสองของเกาส์เซียน เป็นตัวดำเนินการทำงาน ตัวกรองทั้งสองแบบนี้ได้ถูกนำใช้ร่วมกับรูปแบบของปิระมิด และการแปลงเวบเลตแบบมีทิศทาง โดยในขั้นตอนแรกผู้เขียนได้ประเมินทิศทางของภาพลายพิมพ์นิ้วมือ แล้วจำเป็นจะต้องปรับแต่งทิศทางโดยใช้ตัวกรองผ่านความถี่ต่ำ (Low pass filter) ขั้นตอนที่สองได้ใช้ตัวกรองที่มีทิศทางสองแบบคือ

ตัวกรองชนิดการ์บอร์ และตัวกรองอนุพันธ์อันดับสองของเกาส์เซียนเป็นตัวปรับแต่งคุณลักษณะภาพซึ่งเป็นลายเส้นให้ดียิ่งขึ้น ในรูปแบบของการแปลงเวบเลตนั้น ใช้การแปลงเวบเลตแบบมีทิศทางไปในหนึ่งระดับ โดยภาพต้นฉบับจะถูกแบ่งออกเป็นสี่ส่วน คือ ส่วนประมาณค่า, แนวนอน, แนวตั้ง และแนวทแยงมุม ซึ่งขนาดของแต่ละภาคส่วน ได้ทำการลดขนาดลงครึ่งหนึ่ง แล้วจึงใช้ตัวกรองแบบมีทิศทางกับสามส่วนคือ แนวนอน, แนวตั้ง และแนวทแยงมุมก่อน ที่จะประกอบภาพกลับคืนมา ในรูปแบบของปิระมิดนั้น ภาพต้นฉบับจะถูกลดขนาดลงสามระดับ ซึ่งในแต่ละระดับจะใช้ตัวกรองแบบมีทิศทางในการดำเนินการ ก่อนที่ทำการขยายภาพที่ถูกลดของแต่ละดับ หลังนั้นจึงดำเนินการให้ได้ภาพกลับคืน ผลของการศึกษาในครั้งนี้ได้วัดด้วยการทดสอบจุดกึ่งกลางของลายพิมพ์นิ้วมือ โดยใช้วิธีของพอยต์แคร์ และมีฐานข้อมูลที่ใช้ในการศึกษาคือ FVC-2004 การปรับปรุงภาพลายพิมพ์นิ้วมือที่ดีคือจะต้องตรวจจับจุดกึ่งกลางลายพิมพ์นิ้วมือได้ถูกต้องอยู่ในเกณฑ์ที่ดี

Thesis Title	Fingerprint Images Enhancement with Directional filtering
Student	Mr. Keokanlaya Sihalath
Student I.D	5106418
Degree	Master of Engineering
Program	Electronics Engineering
Year	2010
Thesis Advisor	Assoc. Prof. Dr. Somsak Choomchuay

ABSTRACT

Growth in the personal identification by means of biometrics has drawn more attention due to the needs in reliable and easy to use security system with affordable cost. Among the many possible biometric schemes, the fingerprint is one of the most reliable mean for the identification of individual since each person holds distinctive pattern. Not only for personal identification, fingerprints are also widely used in many verification processes. Therefore, the ultimate goal can be assumed to be "fingerprint matching". However to get the speedy and reliability matcher; both fingerprint enhancement and fingerprint classification are unavoidably required. The scanned images are frequently not clear. The unclear or low quality images offer lower recognition rate. Too dry or too wet image are sometimes unavoidable. Unintended printed images (criminal cases) are usually obtained not only in portion but also poor quality. In these circumstances, improvement (or enhancement) of the image is very important and of interest to us.

In this thesis, the author studied only few methods to enhance the quality of fingerprint images. Directional filtering is more focused. Two types similar of filters: Gabor filter and Second derivative of Gaussian filter are investigated. They are also applied in the scheme of pyramid technique and directional wavelet transform. In the process steps, we firstly applied low pass filter in local orientation field estimation for field smoothing. Secondly the Gabor filter or the second derivative of Gaussian filter is applied directly to the image for tuning up the image features. On the scheme of DWT, the directional wavelets transform is investigated at only one level; the original image is divided into four parts: approximation, horizontal, vertical and diagonal, where the size of each part is reduced by a downsampling factor of two. Directional filtering is applied directly those sub images before the reconstruction phase. On the scheme of

the pyramid technique, the original fingerprint is divided into three smaller level images and at each level the directional filtering was applied. The expansion of the reduced fingerprint images is employed in the reconstruction phase. In this study the performance of enhancement is measured by testing the success of core point identification where Poincare technique is used. The commonly well-accepted database FVC-2004 is used in this study. The enhanced fingerprint images offer clean visualization as well as the increase in success true core point detection.

ACKNOWLEDGMENTS

This thesis could not have been written without the support from supervisor. Also I would like to gratitude ASEAN University Network/Southeast Asia Engineering Education Development Network (AUN/SEED-Net) for awarding me the scholarship with the financial support for my study at King Mongkut's Institute of Technology Ludkrabang (KMITL), Thailand.

Firstly, I would like to express my great gratitude towards my supervisor, Assoc. Prof. Dr. Somsak Choomchuay who has given me much suggestion, support, and advice from the very early stage of this research as well as giving me extraordinary experiences throughout the work.

I would also like to gratitude and sincere goes also to all Professors, lectures and supporting staffs in department of electronics engineering, who always continuously encouraged, helped and gave me a guideline during the whole period of my study at KMITL.

I would like to recognize all friends in Thailand who always encouraged and helped me during period of my study. Throughout my years at KMITL, they have enriched my life with their companions and friendship.

Finally, I would also like to my parent for their patience, continuous encouragement, love guidance, and support.

Table of Contents

List of Tables

List of Figures

Figure Page

Chapter 1

Introduction

1.1 Introduction [1]

More than a century has passed since Alphonse Bertillon first conceived and then industriously practiced the idea of using body measurements for solving crimes. Just as his idea was gaining popularity, it faded into relative obscurity by a far more significant and practical discovery of the distinctiveness of the human fingerprints. In 1893, the Home Ministry Office, UK, accepted that no two individuals have the same fingerprints. Soon after this discovery, many major law enforcement departments saw potential of fingerprints in identifying repeat offenders who used an alias, e.g., changed their names with each arrest to evade the harshest penalties reserved for recidivists in law. The law enforcement departments embraced the idea of "booking" the fingerprints of criminals at the time of arrest, so that their records are readily available for later identification. Fingerprints found an application in forensics. By matching leftover fingerprint smudges (latent) from crime scenes to fingerprints collected during booking, authorities could determine the identity of criminals who have been previously arrested. The law enforcement agencies sponsored a rigorous study of fingerprints, developed scientific methods for visual identification/matching of fingerprints and instituted strong programs/ cultures for training fingerprint experts. They successfully applied the art of fingerprint recognition for nailing down the perpetrators.

Fingerprint recognition have been used for a long time to determine person identify in law enforcement, and it has become a more and more important task with increasing demand for person identification in various applications such as online banking, e-commerce, and security control, because fingerprints have significant characteristics such as exchangeability and uniqueness. A fingerprint consists of ridges separated by valleys, these ridges flow almost parallel to each other, and ridges change their flow at minutiae such as endings and bifurcations of ridges. The relation graph between fingerprint minutiae is unique for each finger and does not change during life. Although various algorithms and systems have been proposed as means of determining the identity of fingerprints, most fingerprint identification systems adopt a matching

based on comparison of fingerprint minutiae. However, automatic detection of minutiae tends to be influenced by various noises as caused by dry skin and imprint condition. In order to suppress the influence of noise and enhance signal components of the print, fingerprint enhancement is performed based on fingerprint image properties. Ridge flow is characterized by local ridge direction and local ridge width. The directional features represent an outline of ridge flow. The ridge width between fingerprint ridges is different according to its location, imprint condition, and finger size. As a consequence, these differences result in a variety of ridge frequencies in the fingerprint. Therefore, these features are important to direction filters and to process fingerprint enhancement according to the local ridge features of the print [2].

Currently the most widely used and the most accurate automatic fingerprint identification techniques use minutiae-based automatic fingerprint matching algorithms. Reliably extracting minutiae from the input fingerprint images is critical to fingerprint matching. The performance of current minutiae extraction algorithms depends heavily on the quality of input fingerprint images. In an ideal fingerprint image, ridges and valleys alternate and flow in a locally constant direction and minutiae are anomalies of ridges, that is, ridge endings and ridge bifurcations. In such situations, the ridges can be easily detected and minutiae can be precisely located from a binary ridge map. In practice, due to variations in impression conditions, ridge configuration, skin conditions (dryness, wet finger, aberrant formations of epidermal ridges of fingerprints, postnatal marks, occupational marks), acquisition devices, and non-cooperative attitude of subjects, etc., a significant percentage of acquired fingerprint images from the FVC 2004 DB1_A (approximately 50% according to our experience see in chapter 5) is of poor quality. The ridge structures in poor-quality fingerprint images are not always well defined and hence they cannot always be correctly detected. This, of course, can result in failures of minutiae extraction algorithms, including: a significant number of spurious minutiae may be created, a large percentage of genuine minutiae may be undetected, and a significant amount of error in position and orientation may be introduced. We can see that the performance of the minutiae extraction algorithm on the poor quality image is far from desirable; a significant number of spurious minutiae are created and a large percentage of genuine minutiae are undetected by the algorithm [3].

In order to ensure that the performance of the minutiae extraction algorithms are robust with respect to the quality of input fingerprint images, an enhancement algorithm, which can improve the quality of the ridge structures of input fingerprint images, is thus necessary. Ideally,

when the ridge structures in a fingerprint image are well defined, each ridge is separated by two parallel narrow furrows, each furrow is separated by two parallel narrow ridges; and minutiae are anomalies of ridges, that is, ridge endings and ridge bifurcations. When a fingerprint image is corrupted by various kinds of noise, such well-defined ridge structures are no longer visible. However, a fingerprint expert is often able to correctly identify the minutiae by using various contextual visual clues such as local ridge orientation, ridge continuity, ridge tendency, *etc.*, as long as the ridge and valley structures are not corrupted completely. Therefore, it should be possible to develop an enhancement algorithm that exploits these visual clues to improve the clarity of ridge structures in corrupted fingerprint images. Generally, for a given fingerprint image, fingerprint regions can be assigned as one the following three categories: *i)* well-defined regions, in which ridges and valley are clearly visible for a minutia extraction algorithm to operate reliably, *ii)* recoverable corrupted regions, in which ridges and furrows are corrupted by a small amount of creases, smudges, etc. But they can still be correctly recovered by an enhancement algorithm, and *iii)* unrecoverable corrupted regions, in which ridges and furrows are corrupted by such a severe amount of noise and distortion that it is impossible to recover them from the corrupted image.

We have referred to the first two categories of fingerprint regions as recoverable and the last category as unrecoverable. It is impossible to recover the original ridge structures in the unrecoverable regions, since no ridges and valley are present at all within these regions. Any effort to improve the quality of the fingerprint image in these regions is futile. Therefore, the goal of a reasonable enhancement algorithm is to improve the clarity of ridge structures of fingerprint images in recoverable regions and to mask out the unrecoverable regions. In addition, since the objective of a fingerprint enhancement algorithm is to improve the quality of ridge structures of input fingerprint images to facilitate the extraction of ridges and minutiae, a fingerprint enhancement algorithm should not create any spurious ridge structures. This is very important, because spurious ridge structures may change the individuality of input fingerprints.

Fingerprint enhancement can be performed at either the binary level or the gray level. A binary-level ridge image is an image where all the ridge pixels are assigned a value 1 and non-ridge pixels are assigned a value 0. The binary image can be obtained by applying a ridge extraction algorithm on a gray-level image. Since ridges and valleys in a fingerprint image alternate and run parallel to each other in a local neighborhood, a number of simple heuristics can

be used to differentiate the spurious ridge configurations from the true ridge configurations in a binary ridge image. However, after applying a ridge extraction algorithm on the original gray-level images, information about the true ridge structures is often lost depending on the performance of the ridge extraction algorithm. Therefore, enhancement of binary ridge images has its inherent limitations. In gray-level fingerprint images, ridges and valleys in a local neighborhood form a sinusoidal-shaped plane wave, which has a well-defined frequency and orientation. A number of techniques that take advantage of this information have been proposed to enhance gray-level fingerprint images [4]. However, these algorithms usually assume that the local ridge orientations can be reliably estimated. For fingerprint images of poor quality, such an assumption cannot be made, due to the existence of noise, creases, smudges, and holes. Therefore, a fingerprint enhancement algorithm should not assume that local ridge orientation could be easily obtained. Instead, it should focus a significant amount of effort on reliable estimation of orientation field.

However, most approaches proceeded with the enhancement based on only directional features (minutiae) and do not pay much attention to the ridge frequency, though it is as important as ridge direction. Since the ridge pattern is a cyclical pattern in a local scope, the directional filtering is suitable to describe the print. Based on the intrinsic characteristics of fingerprint patterns, we propose fingerprint images enhancement with directional filtering.

1.2 Biometric Recognition

As our society has become electronically connected and more mobile, surrogate representations of identity such as passwords (prevalent in electronic access control) and cards (prevalent in banking and government applications) cannot be trusted to establish a person's identity. Cards can be lost or stolen and passwords or PIN can, in most cases, be guessed. Further, passwords and cards can be easily shared and so they do not provide non-repudiation [1].

Biometric recognition (or simply biometrics) refers to the use of distinctive anatomical (or physiological) (i.e., fingerprints, face, iris) and behavioral (i.e., speech) characteristics, called biometric identifiers or traits or characteristics for automatically recognizing individuals. Biometrics is becoming an essential component of effective person identification solutions because biometric identifiers cannot be shared or misplaced, and they intrinsically represent the individual's bodily identity. Recognition of a person by their body, then linking that body to an

externally established "identity", forms a very powerful tool of identity management with tremendous potential consequences, both positive and negative. Consequently, biometrics is not only a fascinating pattern recognition research problem but, if carefully used, is an enabling technology with the potential to make our society safer, reduce fraud and provide user convenience (user friendly man–machine interface).

The word biometrics is derived from the Greek words bios (meaning life) and metron (meaning measurement); biometric identifiers are measurements from living human body. Perhaps all biometric identifiers are a combination of anatomical and behavioral characteristics and they should not be exclusively classified into either anatomical or behavioral characteristics. For example, fingerprints are anatomical in nature but the usage of the input device (e.g., how a user presents a finger to the fingerprint scanner) depends on the person's behavior. Thus, the input to the recognition engine is a combination of anatomical and behavioral characteristics. Similarly, speech is partly determined by the vocal tract that produces speech and partly by the way a person speaks. Often, a similarity can be noticed among parents, children, and siblings in their speech. The same argument applies to the face: faces of identical twins may be extremely similar at birth but during their growth and development, the faces change based on the person's behavior (e.g., lifestyle differences leading to a difference in bodyweight, etc.).

A number of questions related to a person's identity are asked everyday in a variety of contexts. Is this person authorized to enter the facility? Is this individual entitled to access privileged information? Is this person wanted for a crime? Has this person already received certain benefits? Is the given service being administered exclusively to the enrolled users? Reliable answers to questions such as these are needed by business and government organizations. Because biometric identifiers cannot be easily misplaced, forged, or shared, they are considered more reliable for person recognition than traditional token (ID cards) or knowledge-based (passwords or PIN) methods. The objectives of biometric recognition are user convenience (e.g., money withdrawal at an ATM machine without a card or PIN), better security (e.g., only authorized person can enter a facility), better accountability (e.g., difficult to deny having accessed confidential records), and higher efficiency (e.g, lower overhead than computer password maintenance). The tremendous success of fingerprint-based recognition technology in law enforcement applications, decreasing cost of fingerprint sensing devices, increasing availability of inexpensive computing power, and growing identity fraud/theft have all resulted in

increasing use of fingerprint-based person recognition in commercial, government, civilian, and financial domains. In addition to fingerprints, some other traits, primarily hand shape, voice, iris and face have also been successfully deployed.

1.3 Biometric Systems

An important issue in designing a practical biometric system is to determine how an individual is going to be recognized. Depending on the application context, a biometric system may be called either a verification system or an identification system [1]:

- A verification system authenticates a person's identity by comparing the captured biometric characteristic with her previously captured (enrolled) biometric reference template pre-stored in the system. It conducts one-to-one comparison to confirm whether the claim of identity by the individual is true. A verification system either rejects or accepts the submitted claim of identity.
- An identification system recognizes an individual by searching the entire enrollment template database for a match. It conducts one-to-many comparisons to establish if the individual is present in the database and if so, returns the identifier of the enrollment reference that matched. In an identification system, the system establishes a subject's identity (or determines that the subject is not enrolled in the system database) without the subject having to claim an identity.

The term authentication is also used in the biometric field, sometimes as a synonym for verification; actually, in the information technology language, authenticating a user means to let the system know the identity of the user regardless of the mode (verification or identification). Throughout this book we use the generic term recognition where we are not interested in distinguishing between verification and identification.

The block diagrams of verification and identification systems are depicted in Figure 1.1 user enrollment, which is common to both tasks is also graphically illustrated.

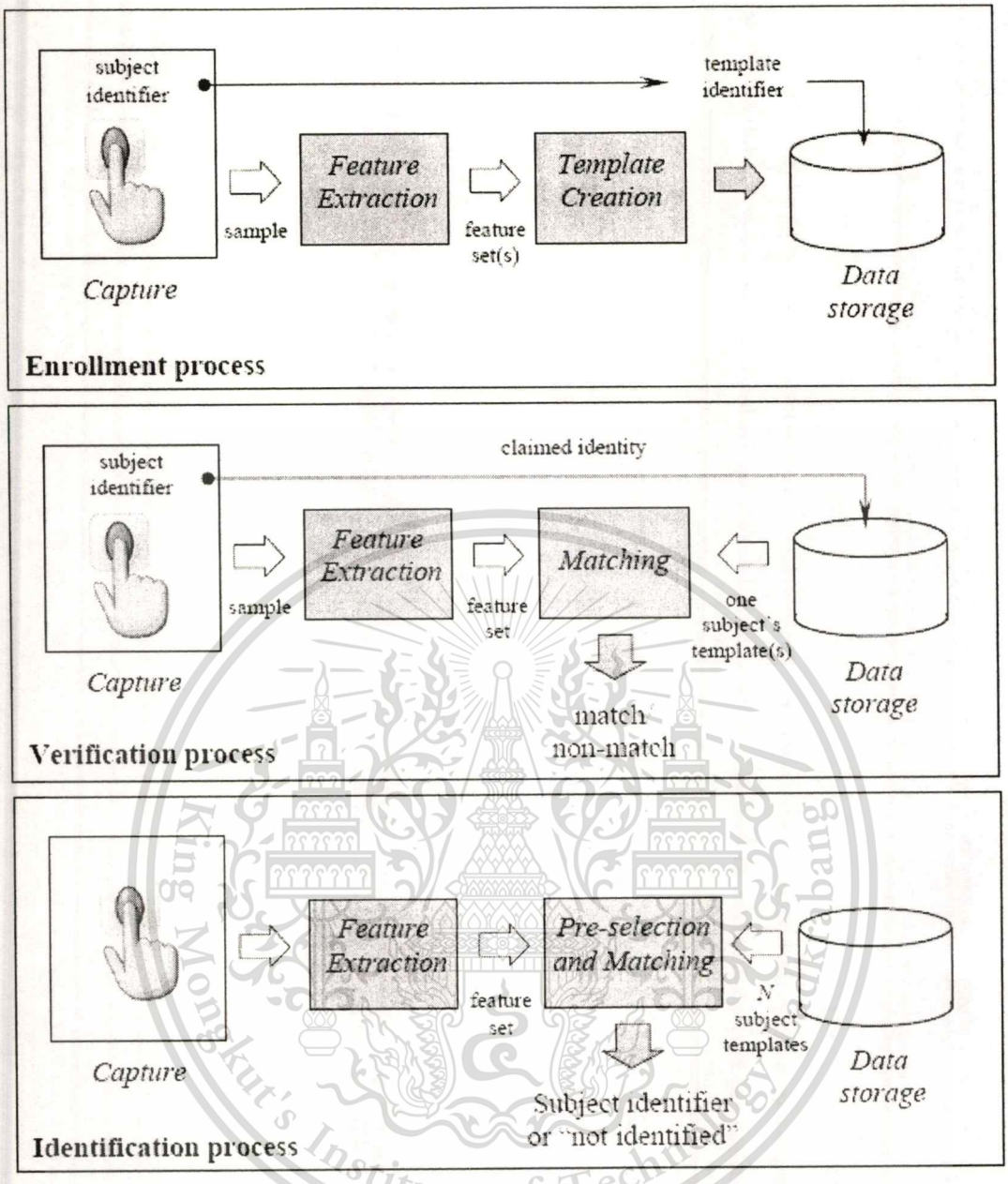


Figure 1.1 Enrollment, verification, and identification processes, these processes use the following modules.

Capture, feature extraction, template creation, matching, pre-selection, and data storage. In the identification process pre-selection and matching are often combined [1]. The enrollment, verification, and identification processes involved in user recognition make use of the following system modules:

- Capture: a digital representation of biometric characteristic needs to be sensed and captured. A biometric sensor, such as a fingerprint scanner, is one of the central pieces of a biometric capture module. The captured digital representation of the

biometric characteristic is often known as a *sample*; for example, in the case of a fingerprint system, the raw digital fingerprint image captured by the fingerprint scanner is the sample. The data capture module may also contain other components (e.g., a keyboard and screen) to capture other (non-biometric) data.

- Feature extraction: in order to facilitate matching or comparison, the raw digital representation (sample) is usually further processed by a feature extractor to generate a compact but expressive representation, called a feature set.
- Template creation: the template creation module organizes one or more feature sets into an enrollment template that will be saved in some persistent storage. The enrollment template is sometimes also referred to as a reference.
- Pre-selection and matching: the pre-selection (or filtering) stage is primarily used in an identification system when the number of enrolled templates is large. Its role is to reduce the effective size of the template database so that the input needs to be matched to a relatively small number of templates. The matching (or comparison) stage (also known as a matcher) takes a feature set and an enrollment template as inputs and computes the similarity between them in terms of a matching score, also known as similarity score. The matching score is compared to a system threshold to make the final decision; if the match score is higher than the threshold, the person is recognized, otherwise not.
- Data storage: is devoted to storing templates and other demographic information about the user. Depending on the application, the template may be stored in internal or external storage devices or be recorded on a smart card issued to the individual.

Using these five modules, three main processes can be performed, namely, enrollment, verification, and identification. A verification system uses the enrollment and verification processes while an identification system uses the enrollment and identification processes. The three processes are:

- Enrollment: user enrollment is a process that is responsible for registering individuals in the biometric system storage. During the enrollment process, the biometric characteristic of a subject is first captured by a biometric scanner to produce a sample. A quality check is often performed to ensure that the acquired sample can be reliably processed by successive stages. A feature extraction module

is then used to produce a feature set. The template creation module uses the feature set to produce an enrollment template. Some systems collect multiple samples of a user and then either select the best image (or feature set) or fuse multiple images (or feature sets) to create a compos ite template. The enrollment process then takes the enrollment template and stores it in the system storage together with the demographic information about the user (such as an identifier, name, gender, height, etc.).

- Verification: the verification process is responsible for confirming the claim of identity of the subject. During the recognition phase, an identifier of the subject (such as username or PIN [Personal Identification Number]) is provided (e.g., through a keyboard or a keypad or a proximity card) to claim an identity; the biometric scanner captures the characteristic of the subject and converts it to a sample, which is further processed by the feature extraction module to produce a feature set. The resulting feature set is fed to the matcher, where it is compared against the enrollment template(s) of that subject (retrieved from the system storage based on the subject's identifier). The verification process produces a match/non-match decision.
- Identification: in the identification process, the subject does not explicitly claim an identity and the system compares the feature set (extracted from the captured biometric sample) against the templates of all (or a subset of) the subjects in the system storage; the output is a *candidate list* that may be empty (if no match is found) or contain one (or more) identifier(s) of matching enrollment templates. Because identification in large databases is computationally expensive, a pre-selection stage is often used to filter the number of enrollment templates that have to be matched against the input feature set.

Depending on the application domain, a biometric system could operate either as an online system or an off-line system. An on-line system requires the recognition to be performed quickly and an immediate response is imposed (e.g., a computer network logon application). On the other hand, an off-line system does not require the recognition to be performed immediately and a relatively longer response delay is allowed (e.g., background check of an applicant). On-line systems are often fully automatic and require that the biometric characteristic be captured

using a live-scan scanner, the enrollment process be unattended, there be no (manual) quality control, and the matching and decision making be fully automatic. Off-line systems, however, are often semi-automatic, where the biometric acquisition could be through an offline scanner (e.g., scanning a fingerprint image from a latent or inked fingerprint card), the enrollment may be supervised (e.g., when a suspect is "booked," a police officer guides the fingerprint acquisition process), a manual quality check may be performed to ensure good quality acquisition, and the matcher may return a list of candidates which are then manually examined by a forensic expert to arrive at a final decision.

The verification and identification processes differ in whether an identity is claimed or not by the subject. A biometric claim (or claim of identity) is defined as the implicit or explicit claim that a subject is or is not the source of a specified or unspecified biometric enrollment template. A claim may be [1]:

- Positive: the subject is enrolled.
- Negative: the subject is not enrolled.
- Specific: the subject is or is not enrolled as a specified biometric enrollee.
- Non-specific: the subject is or is not among a set or subset of biometric enrollees.

The application context defines the type of claim. In certain applications, it is in the interest of the subject to make a positive claim of identity. Such applications are typically trying to prevent multiple people from using the same identity. For example, if only Alice is authorized to enter a certain secure area, then it is in the interest of any subject to make a positive claim of identity (of being Alice) to gain access. But the system should grant access only to Alice. If the system fails to match the enrolled template of Alice with the input feature set, access is denied, otherwise, access is granted. In other applications, it is in the interest of the subject to make a negative claim of identity. Such applications are typically trying to prevent a single person from using multiple identities. For example, if Alice has already received certain welfare benefits, it is in her interest to now make a negative claim of identity (that she is not among the people who have already received benefits), so that she can double-dip. The system should establish that Alice's negative claim of identity is false by finding a match between the input feature set of Alice and enrollment templates of all people who have already received the benefits. The following three types of claims are used depending on the application context:

- Specific positive claim: applications such as logical access control (e.g., network logon) may require a specific positive claim of identity (e.g., through a username or PIN). A verification biometric system is sufficient in this case to confirm whether the specific claim is true or not through a one-to-one comparison.
- Non-specific positive claim: applications such as physical access control may assume a non-specific positive claim that the subject is someone who is authorized to access the facility. One of the advantages of this scenario is that the subject does not need to make a specific claim of identity (no need to provide a username, PIN, or any other token), which is quite convenient. However, the disadvantage of this scenario is that an identification biometric system is necessary (which has longer response time and lower accuracy due to one-to-many comparisons).
- Non-specific negative claim: applications such as border crossing typically assume a non-specific negative claim, i.e., the subject is not present in a "watch list". Again, an identification system must be used in this scenario. Note that such applications cannot use traditional knowledge-based or possession-based methods of recognition. Surrogates tokens such as passports have been traditionally used in such applications but if passports are forged (or if people obtain duplicate passports under different names), traditional recognition methods cannot solve the problem of duplicate identities or multiple enrollments.

1.4 Biometric Characteristics

Identity verification in computer systems is done based on measures like keys, cards, passwords, PIN and so forth. Unfortunately, these may often be forgotten, disclosed or changed. A reliable and accurate identification/verification technique may be designed using biometric technologies, which are further based on the special characteristics of the person such as face, iris, fingerprint, signature and so forth. This technique of identification is preferred over traditional passwords and PIN-based techniques for various reasons [1, 5]:

- The person to be identified is required to be physically present at the time of identification.
- Identification based on biometric techniques obviates the need to remember a password or carry a token.

There are many biometric characteristics may be captured in the first phase of processing. However, automated capturing and automated comparison with previously stored data requires the following properties of biometric characteristics:

- Universal: Everyone must have the attribute. The attribute must be one that is seldom lost to accident or disease.
- Invariance of properties: They should be constant over a long period of time. The attribute should not be subject to significant differences based on age or either episodic or chronic disease.
- Measurability: The properties should be suitable for capture without waiting time and it must be easy to gather the attribute data passively.
- Singularity: Each expression of the attribute must be unique to the individual. The characteristics should have sufficient unique properties to distinguish one person from any other. Height, weight, hair and eye color are unique attributes, assuming a particularly precise measure, but do not offer enough points of differentiation to be useful for more than categorizing.
- Acceptance: The capturing should be possible in a way acceptable to a large percentage of the population. Excluded are particularly invasive technologies; that is, technologies requiring a part of the human body to be taken or (apparently) impairing the human body.
- Reducibility: The captured data should be capable of being reduced to an easy-to-handle file.
- Reliability and tamper-resistance: The attribute should be impractical to mask or manipulate. The process should ensure high reliability and reproducibility.
- Privacy: The process should not violate the privacy of the person.
- Comparable: The attribute should be able to be reduced to a state that makes it digitally comparable to others. The less probabilistic the matching involved, the more authoritative the identification.
- Inimitable: The attribute must be irreproducible by other means. The less reproducible the attribute, the more likely it will be authoritative.

Among the various biometric technologies being considered are fingerprint, facial features, hand geometry, voice, iris, retina, vein patterns, palm print, DNA, keystroke dynamics, ear shape, odor, signature and so forth [1, 5]:

Fingerprint: fingerprint biometric is an automated digital version of the old ink-and-paper method used for more than a century for identification, primarily by law enforcement agencies. The biometric device requires each user to place a finger on a plate for the print to be read. Fingerprint biometrics currently has three main application areas: large-scale Automated Finger Imaging Systems (AFIS), generally used for law enforcement purposes; fraud prevention in entitlement programs; and physical and computer access. A major advantage of finger imaging is the long-time use of fingerprints and its wide acceptance by the public and law enforcement communities as a reliable means of human recognition. Others include the need for physical contact with the optical scanner, possibility of poor-quality images due to residue on the finger such as dirt and body oils (which can build up on the glass plate), as well as eroded fingerprints from scrapes, years of heavy labor or mutilation.

Figure 1.2 Fingerprint.

Facial recognition: face recognition is a noninvasive process where a portion of the subject's face is photographed and the resulting image is reduced to a digital code. Facial recognition records the spatial geometry of distinguishing features of the face. Facial recognition technologies can encounter performance problems stemming from such factors as non-cooperative behavior of the user, lighting and other environmental variables. The main disadvantages of face recognition are similar to problems of photographs. People who look alike

can fool the scanners. There are many ways in which people can significantly alter their appearance, like slight change in facial hair and style.

Figure 1.3 Face.

Iris scan: iris scanning measures the iris pattern in the colored part of the eye, although iris color has nothing to do with the biometric. Iris patterns are formed randomly. As a result, the iris patterns in the left and right eyes are different, and so are the iris patterns of identical twins. Iris templates are typically around 256 bytes. Iris scanning can be used quickly for both identification and verification applications because of its large number of degrees of freedom. Disadvantages of iris recognition include problems of user acceptance, relative expense of the system as compared to other biometric technologies and the relatively memory-intensive storage requirements.

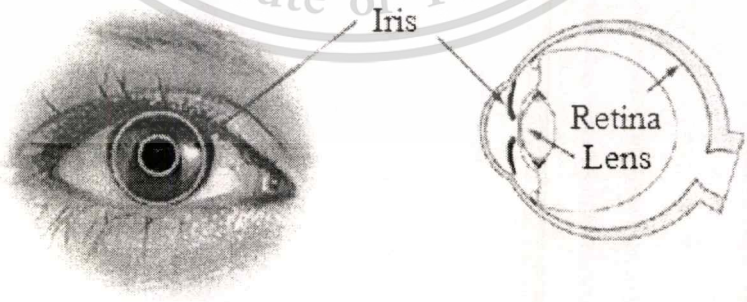


Figure 1.4 Iris scans.

Retinal scan: retinal scanning involves an electronic scan of the retina the innermost layer of wall of the eyeball. By emitting a beam of incandescent light that bounces off the person's retina and returns to the scanner, a retinal scanning system quickly maps the eye's blood vessel pattern and records it into an easily retrievable digitized database. The eye's natural reflective and absorption properties are used to map a specific portion of the retinal vascular structure. The advantages of retinal scanning are its reliance on the unique characteristics of each person's retina, as well as the fact that the retina generally remains fairly stable throughout life. Disadvantages of retinal scanning include the need for fairly close physical contact with the scanning device. Also, trauma to the eye and certain diseases can change the retinal vascular structure, and there also are concerns about public acceptance.

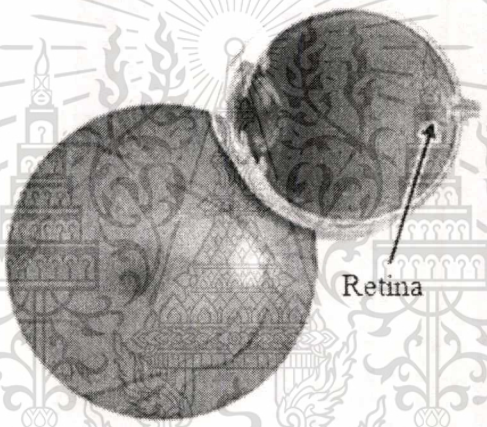


Figure 1.5 Retina.

Voice recognition: voice or speaker recognition uses vocal characteristics to identify individuals using a pass-phrase. It involves taking the acoustic signal of a person's voice and converting it to a unique digital code that can be stored in a template. Voice recognition systems are extremely well-suited for verifying user access over a telephone. Disadvantages of this biometric are that not only is a fairly large byte code required, but also, people's voices can change (for example, when they are sick or in extreme emotional states). Also, phrases can be misspoken and background noises can interfere with the system.

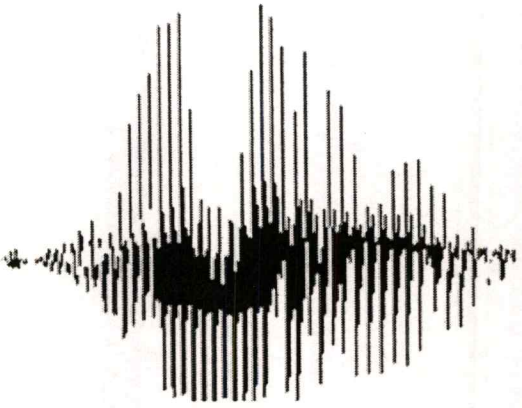


Figure 1.6 Voice.

Signature verification: it is an automated method of examining an individual's signature. This technology examines dynamics such as speed, direction and pressure of writing; the time that the stylus is in and out of contact with the "paper"; the total time taken to make the signature; and where the stylus is raised from and lowered onto the "paper". Signature verification templates are typically 50 to 300 bytes. The key is to differentiate between the parts of the signature that are habitual and those that vary with almost every signing. Disadvantages include problems with long-term reliability, lack of accuracy and cost.

Figure 1.7 Signature.

Hand/Finger geometry: hand or finger geometry is an automated measurement of many dimensions of the hand and fingers. Neither of these methods takes actual prints of palm or fingers. Only the spatial geometry is examined as the user puts a hand on the sensor's surface. Hand geometry templates are typically 9 bytes, and finger geometry templates are 20 to 25 bytes. Finger geometry usually measures two or three fingers, and thus requires a small amount of

computational and storage resources. The problems with this approach are that it has low discriminative power, the size of the required hardware restricts its use in some applications and hand geometry-based systems can be easily circumvented.

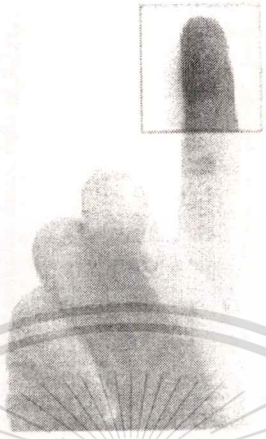


Figure 1.8 Finger.

Palm print: palm print verification is a slightly modified form of fingerprint technology. Palm print scanning uses an optical reader very similar to that used for fingerprint scanning; however, its size is much bigger, which is a limiting factor for use in workstations or mobile devices.

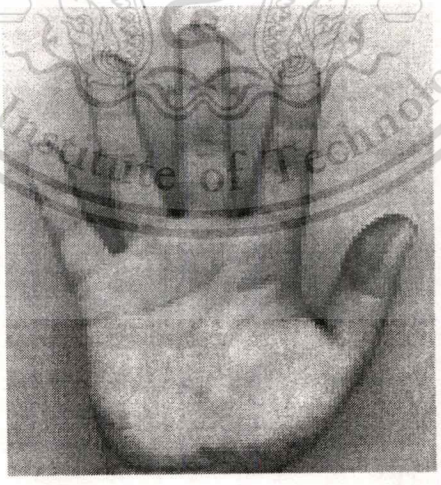


Figure 1.9 Palm print.

Keystroke dynamics: keystroke dynamics is an automated method of examining an individual's keystrokes on a keyboard. This technology examines dynamics such as speed and

pressure, the total time of typing a particular password and the time that a user takes between hitting keys-dwell time (the length of time one holds down each key) as well as flight time (the time it takes to move between keys). Taken over the course of several login sessions, these two metrics produce a measurement of rhythm unique to each user. Technology is still being developed to improve robustness and distinctiveness.

Vein Patterns (hand vein): vein geometry is based on the fact that the vein pattern is distinctive for various individuals. Vein measurement generally focuses on blood vessels on the back of the hand. The veins under the skin absorb infrared light and thus have a darker pattern on the image of the hand. An infrared light combined with a special camera captures an image of the blood vessels in the form of tree patterns. This image is then converted into data and stored in a template. Vein patterns have several advantages: First, they are large, robust internal patterns. Second, the procedure does not implicate the criminal connotations associated with the taking of fingerprints. Third, the patterns are not easily damaged due to gardening or bricklaying. However, the procedure has not yet won full mainstream acceptance. The major disadvantage of vein measurement is the lack of proven reliability.

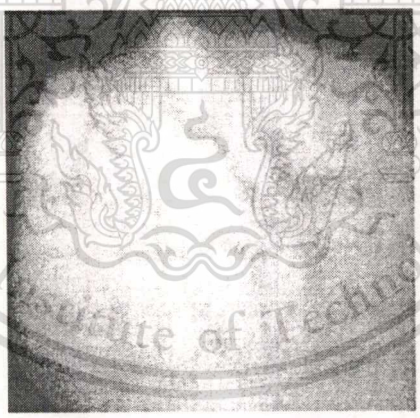


Figure 1.10 Hand vein.

DNA (Deoxyribonucleic Acid): DNA sampling is rather intrusive at present and requires a form of tissue, blood or other bodily sample. This method of capture still has to be refined. So far, DNA analysis has not been sufficiently automatic to rank it as a biometric technology. The analysis of human DNA is now possible within 10 minutes. If the DNA can be matched automatically in real time, it may become more significant. At present, DNA is very entrenched in crime detection and will remain in the law enforcement area for the time being.

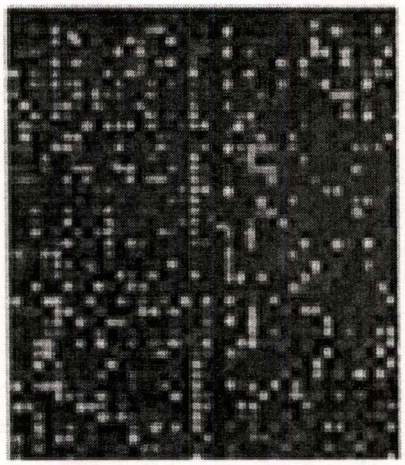


Figure 1.11 DNA.

Ear shape: identifying individuals by ear shape is used in law enforcement applications where ear markings are found at crime scenes. Problems are faced whenever the ear is covered by hair.

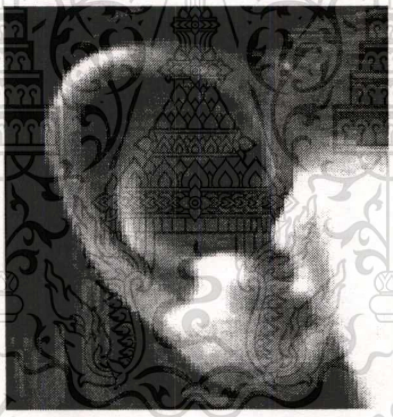


Figure 1.12 Ear.

Body odor: the body odor biometrics is based on the fact that virtually every human's smell is unique. The smell is captured by sensors that are capable of obtaining the odor from non-intrusive parts of the body, such as the back of the hand. The scientific basis is that the chemical composition of odors can be identified using special sensors. Each human smell is made up of chemicals known as volatiles. They are extracted by the system and converted into a template. The use of body odor sensors broaches on the privacy issue, as the body odor carries a significant amount of sensitive personal information. It is possible to diagnose some disease or activities in last hours by analyzing body odor.

The biometric identifiers described above are compared in Table 1.1. Note that fingerprint has a nice balance among all the desirable properties. Every human being possesses fingers (with the exception of hand-related disability) and hence fingerprints. Fingerprints are very distinctive and they are permanent; even if they temporarily change slightly due to cuts and bruises on the skin, the fingerprint reappears after the finger heals. Live-scan fingerprint scanners can easily capture high-quality fingerprint images and unlike face recognition, they do not suffer from the problem of segmenting the fingerprint from the background. However, they are not suitable for covert applications (e.g., surveillance) as live-scan fingerprint scanners cannot capture a fingerprint image from a distance and without the knowledge of the person. The deployed fingerprint identification (recognition) systems offer good performance and fingerprint scanners have become quite compact and affordable. Because fingerprints have a long history of use in forensic divisions worldwide for criminal investigations, they have some stigma of criminality associated with them. However, this is rapidly changing with the high demand for automatic person recognition to fight identity fraud and security threats. With a layered approach involving fingerprint and other security technologies, fingerprint systems are difficult to circumvent. Fingerprint enhancement based recognition is one of the most mature biometric technologies and is suitable for a large number of recognition applications. This is also reflected in the revenues generated by various biometric technologies.

Table 1.1 Comparison of commonly used biometric characteristics.

Biometric identifier	Universality	Distinctiveness	Permanence	Collectability	Performance	Acceptability	Circumvention
Face	H	L	M	H	L	H	H
Fingerprint	M	H	H	M	H	M	M
Hand geometry	M	M	M	H	M	M	M
Hand/finger vein	M	M	M	M	M	M	L
Iris	H	H	H	M	H	L	L
Signature	L	L	L	H	L	H	H
Voice	M	L	L	M	L	H	H

Entries in the table are based on the perception of the authors. High, Medium, and Low are denoted by H, M, and L, respectively.

1.5 Applications of Fingerprint Systems

Fingerprint recognition systems have been deployed in a wide variety of application domains, ranging from forensics to mobile phones. But, the system design depends on the application characteristics that define the application requirements. Fingerprint one of most biometrics has been widely used in forensics applications such as criminal identification and prison security. The biometric technology is rapidly evolving and has a very strong potential to be widely adopted in civilian applications such as electronic banking, e-commerce, and access control (see Figure 1.13). Due to a rapid increase in the number and use of electronic transactions, electronic banking and electronic commerce are becoming one of the most important emerging applications of biometrics. These applications include credit card and smart card security, ATM security, check cashing and fund transfers, online transactions and web access. The physical access control applications have traditionally used token-based authentication. With the progress in biometric technology, these applications will increasingly use biometrics for authentication. Remote login and data access applications have traditionally used knowledge-based authentication. These applications have already started using biometrics for person authentication. The use of biometrics will become more widespread in coming years as the technology matures and becomes more trust worthy. Other biometric applications include welfare disbursement, immigration checkpoints, national ID, voter and driver registration, and time and attendance.

The two most popular ways to categorize fingerprint-based biometric recognition applications are horizontal categorization and vertical categorization. In horizontal categorization, the categories are applications that have some commonalties in the features that they require from the fingerprint recognition system. The vertical categorization is based on the needs of a particular sector of industry or the government. Horizontal categorization results in the following main categories of biometric applications [1]:

- Physical access control: access is restricted to facilities such as nuclear plants, bank vaults, corporate board rooms, and even health-clubs, amusement parks, and lockers.
- Logical access control: access to desktop computers or remote servers and databases is restricted to authorized users. Increasingly, access to software applications is also being restricted to only authorize users.
- Transaction authentication (or consumer identification): transactions may be executed at ATM site or from remote locations for on-line banking or between banks

(i.e., in high-value transactions). Fingerprint recognition systems are used for security of the transaction as well as accountability (so the parties involved in the transaction cannot later deny it).

- Device access control: laptops, PDAs, cell phones, and other electronic devices often contain personal and sensitive data. To protect such data, fingerprint recognition systems are used to conduct recognition on the stand-alone device.
- Time and attendance: time and attendance systems are used to keep track of employee working hours and to compute payrolls. Use of fingerprint recognition systems in these applications is fairly well received to improve efficiency for employees and also for preventing various types of payroll frauds (e.g., buddy-punching).
- Civil identification: in civilian identification application, the most important objective is to prevent multiple enrollments and to find duplicates (e.g., duplicate passport, driver license, national identification card). The size of the database can be of the order of millions (e.g., the entire population of a country). In some applications (such as border control to prevent suspected terrorists or expellees from entering the country), the identification is not needed to be conducted against the entire population but rather against a "watch-list" database.
- Forensic identification: in forensic identification, latent fingerprints lifted from the crime scenes are matched against a criminal database to identify the suspect (and sometimes the victims).

Vertical categorization results in the following main industries that benefit the most from the use of fingerprint systems:

- Health care
- Financial Gaming and hospitality (casinos, hotels, etc:)
- Retail
- Education
- Manufacturing
- High technology and telecommunications
- Travel and transport
- Federal, state, municipal, or other governments

- Military
- Law enforcement

Each vertical market may have a need for a number of different horizontal applications. For example, while the most widespread (almost ubiquitous) use of fingerprint identification systems in law enforcement departments is for criminal investigations, these departments also use computers that contain sensitive data. So, this sector needs solutions for fingerprint-based logical access control. Further, law enforcement departments have laboratories and other restricted physical areas, so they can benefit from fingerprint-based physical access control solutions. Fingerprint-based time and attendance solutions can also be used to manage payroll of law enforcement officers (and other employees of the department).

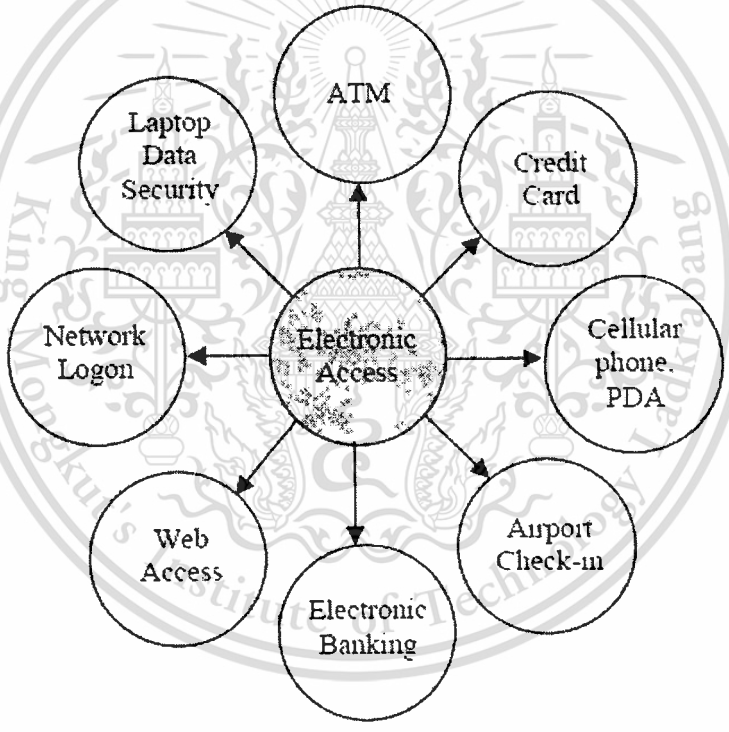


Figure 1.13 Various electronic access applications in widespread use that require automatic authentication.

1.6 History of Fingerprints [1]

Human fingerprints have been discovered on a large number of archaeological artifacts and historical items (see Figure 1.14). While these findings provide evidence that ancient people were aware of the individuality of fingerprints, such awareness does not appear to have any scientific basis. It was not until the late sixteenth century that the modern scientific fingerprint technique was first initiated. In 1864, the English plant morphologist, Nehemiah Grew, published the first scientific paper reporting his systematic study on the ridge, valley, and pore structure in fingerprints Figure1.15 a).

The first detailed description of the anatomical formation of fingerprints was made by Mayer in 1788 in which a number of fingerprint ridge characteristics were identified and characterized Figure1.15 b). Starting in 1809, Thomas Bewick started using fingerprint as his trademark Figure 1.16 a), one of the most important milestones in the history of fingerprints. Purkinje, in 1823, proposed the first fingerprint classification scheme, which classified fingerprints into nine categories according to the ridge configurations Figure 1.16 b)

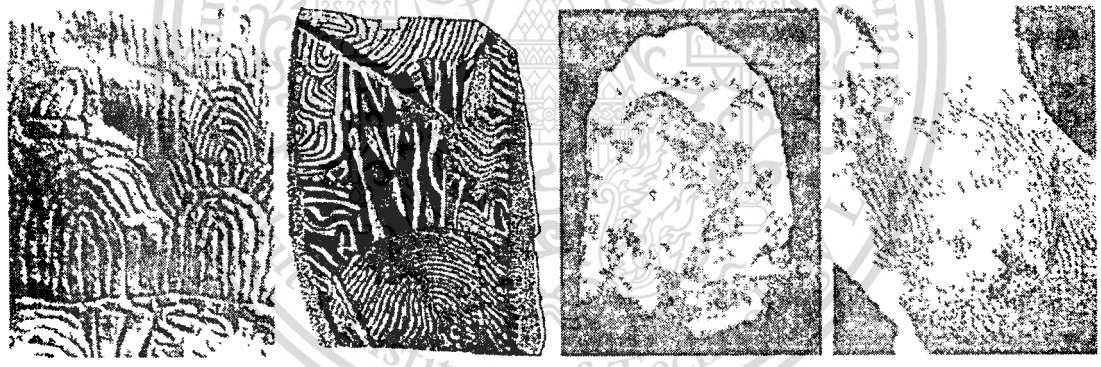


Figure 1.14 Examples of archaeological fingerprint carvings and historic fingerprint impressions.

a) Neolithic carvings (Gavrinis Island); b) standing stone (Goat Island), c) a Chinese clay seal and d) an impression on a Palestinian lamp. While impressions on the Neolithic carvings and the Goat Island standing stones might not be used to establish identity, there is sufficient evidence to suggest that the Chinese clay seal and impressions on the Palestinian lamp were used to indicate the identity of the fingerprint providers.

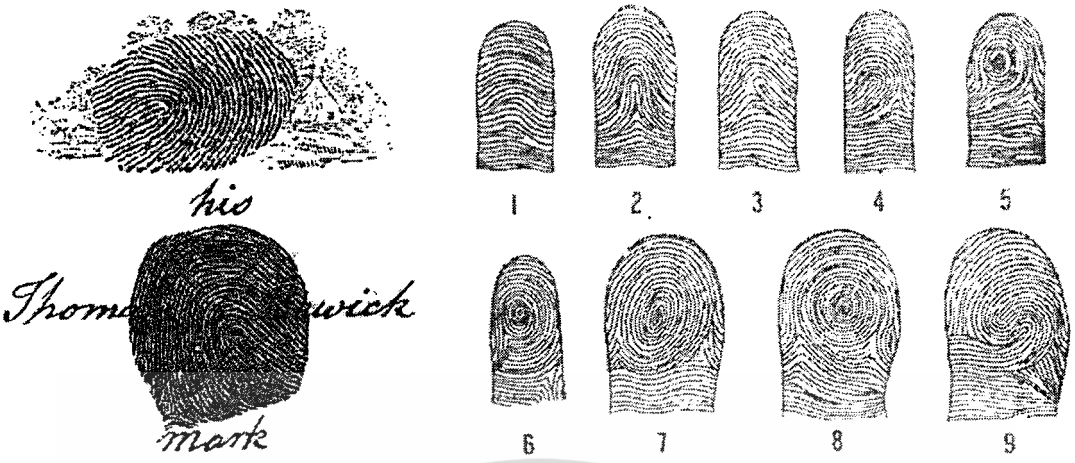
Henry Fauld, in 1880, first scientifically suggested the individuality of fingerprints based on empirical observations. At the same time, Herschel asserted that he had practiced fingerprint recognition for about 20 years. These findings established the foundation of modern fingerprint recognition. In the late nineteenth century, Sir Francis Galton conducted an extensive study on fingerprints. He introduced the minutiae features for comparing fingerprints in 1888.

An important advance in fingerprint recognition was made in 1899 by Edward Henry, who (actually his two assistants from India) established the well-known "Henry system" of fingerprint classification. By the early twentieth century, the formation of fingerprints was well understood. The biological principles of fingerprints are summarized below:

1. Individual epidermal ridges and valley have different characteristics for different fingerprints.
2. The configuration types are individually variable, but they vary within limits that allow for a systematic classification.
3. The configurations and minute details of individual ridges and furrows are permanent and unchanging.

a) Dermatoglyphics drawn by Grew. b) Mayer's drawings of fingerprints.

Figure 1.15 Example of the first scientific proposed ridge, valley and pore structure in fingerprint.



a) trademark of Thomas Bewick; b) the nine patterns illustrated in Purkinje's thesis

Figure 1.16 Example of first using fingerprint and first classification scheme.

The first principle constitutes the foundation of fingerprint recognition and the second principle constitutes the foundation of fingerprint classification. In the early twentieth century, fingerprint recognition was formally accepted as a valid personal identification method and became a standard routine in forensics. Fingerprint identification agencies were set up worldwide and criminal fingerprint databases were established. Various fingerprint recognition techniques, including latent fingerprint acquisition, fingerprint classification, and fingerprint comparison were developed. For example, the FBI fingerprint identification division was set up in 1924 with a database of 810,000 fingerprint cards (see Federal Bureau of Investigation).

With the rapid expansion of fingerprint recognition in forensics, operational fingerprint databases became so huge that manual fingerprint identification became infeasible. For example, the total number of fingerprint cards (each card contains one impression for each of the 10 fingers of a person) in the FBI fingerprint database now stands well over 200 million from its original number of 810,000 and is growing continuously. With thousands of requests being received daily, even a team of more than 1,300 fingerprint experts were not able to provide timely responses to these requests. Starting in the early 1960s, the FBI, Home Office in the UK, and Paris Police Department began to invest a large amount of effort in developing automatic fingerprint identification systems. Based on the observations of how human fingerprint experts perform fingerprint recognition, three major problems in designing AFISs were identified and investigated: digital fingerprint acquisition, local ridge characteristic extraction, and ridge

characteristic pattern matching. Their efforts were so successful that today almost every law enforcement agency worldwide uses an AFIS. These systems have greatly improved the operational productivity of law enforcement agencies and reduced the cost of hiring and training human fingerprint experts.

Fingerprint enhancement based automatic fingerprint recognition technology has now rapidly grown beyond forensic applications into civilian and commercial applications. In fact, fingerprint-based biometric systems are so popular that they have almost become the synonym for biometric systems.

1.7 Formation of Fingerprints

Fingerprints are fully formed at about seven months of fetus development and finger ridge configurations do not change throughout the life of an individual except due to accidents such as bruises and cuts on the finger tips. This property makes fingerprints a very attractive biometric identifier. Biological organisms, in general, are the consequence of the interaction of genes and environment. It is assumed that the phenotype is uniquely determined by the interaction of a specific genotype and a specific environment. Physical appearance and fingerprints are, in general, a part of an individual's phenotype. In the case of fingerprints, the genes determine the general characteristics of the pattern. Fingerprint formation is similar to the growth of capillaries and blood vessels in angiogenesis. The general characteristics of the fingerprint emerge as the skin on the fingertip begins to differentiate. However, the flow of amniotic fluids around the fetus and its position in the uterus change during the differentiation process. Thus, the cells on the fingertip grow in a microenvironment that is slightly different from hand to hand and finger to finger. The finer details of the fingerprints are determined by this changing microenvironment. A small difference in microenvironment is amplified by the differentiation process of the cells. There are so many variations during the formation of fingerprints that it would be virtually impossible for two fingerprints to be alike. But since the fingerprints are differentiated from the same genes, they will not be totally random patterns either. We could say that the fingerprint formation process is a chaotic system rather than a random one.

1.8 Individuality of Fingerprints

Although the word "fingerprint" is popularly perceived as synonymous with individuality, uniqueness of fingerprints is not an established fact but an empirical observation. With the stipulation of widespread use of fingerprints, however, there is a rightfully growing public concern about the scientific basis underlying individuality of fingerprints. Lending erroneous legitimacy to these observations will have disastrous consequences, especially if fingerprints will be ubiquitously used to recognize citizens for reasons of efficiency, convenience, and reliability in guarding against security threats and identity fraud. Furthermore, automated fingerprint recognition systems do not appear to use all the available discriminatory information in the fingerprints, but only a parsimonious representation extracted by an automatic feature extraction algorithm.

1.9 Fingerprint Sensing and Storage [1]

Based on the mode of acquisition, a fingerprint image may be classified as off-line or live-scan. An off-line image is typically obtained by smearing ink on the fingertip and creating an inked impression of the fingertip on paper. The inked impression is then digitized by scanning the paper using an optical scanner or a high-quality video camera. A live-scan image, on the other hand, is acquired by sensing the tip of the finger directly, using a sensor that is capable of digitizing the fingerprint on contact. Particular kind of off-line images, extremely important in forensic applications, are the so-called *latent* fingerprints found at crime scenes. The oily nature of the skin results in the impression of a fingerprint being deposited on a surface that is touched by a finger. These latent prints can be "lifted" from the surface by employing certain chemical techniques.

The main parameters characterizing a digital fingerprint image are: resolution, area, number of pixels, geometric accuracy, contrast, and geometric distortion. To maximize compatibility between digital fingerprint images and to ensure good quality of the acquired fingerprint impressions among various AFIS, the US Criminal Justice Information Services (CJIS) released a set of specifications that regulate the quality and the format of both fingerprint images and FBI-compliant off-line/live-scan scanners. More recently, FBI has defined another, less stringent, image quality standard for single-finger capture devices in civilian applications (more specifically for the Personal Identity Verification [PIV] program in the United States).

Most of the commercial live-scan devices, designed for the non-AFIS market, do not meet the FBI specifications but, on the other hand, are designed to be compact, and cheap. The operational quality of fingerprint scanners (e.g., the impact of the scanner quality parameters on the fingerprint recognition accuracy) has been the subject of some recent studies.

There are a number of live-scan sensing mechanisms (e.g., optical FTIR, capacitive, thermal, pressure-based, ultrasound, etc.) that can be used to detect the ridges and valleys present on the fingertip. Figure 1.17 shows an off-line fingerprint image acquired with the ink technique, a latent fingerprint image, and some live-scan images acquired with different types of commercial live-scan devices. Although optical fingerprint scanners have the longest history, the new solid-state sensors are gaining increasing popularity because of their compact size and the ease with which they can be embedded in consumer products such as laptop computers, cellular phones and PDAs. Figure 1.17 fingerprint images from a) a live-scan FTIR-based optical scanner; b) a live-scan capacitive scanner; c) a live-scan piezoelectric scanner; d) a live-scan thermal scanner; e) an off-line inked impression, and f) a latent fingerprint.

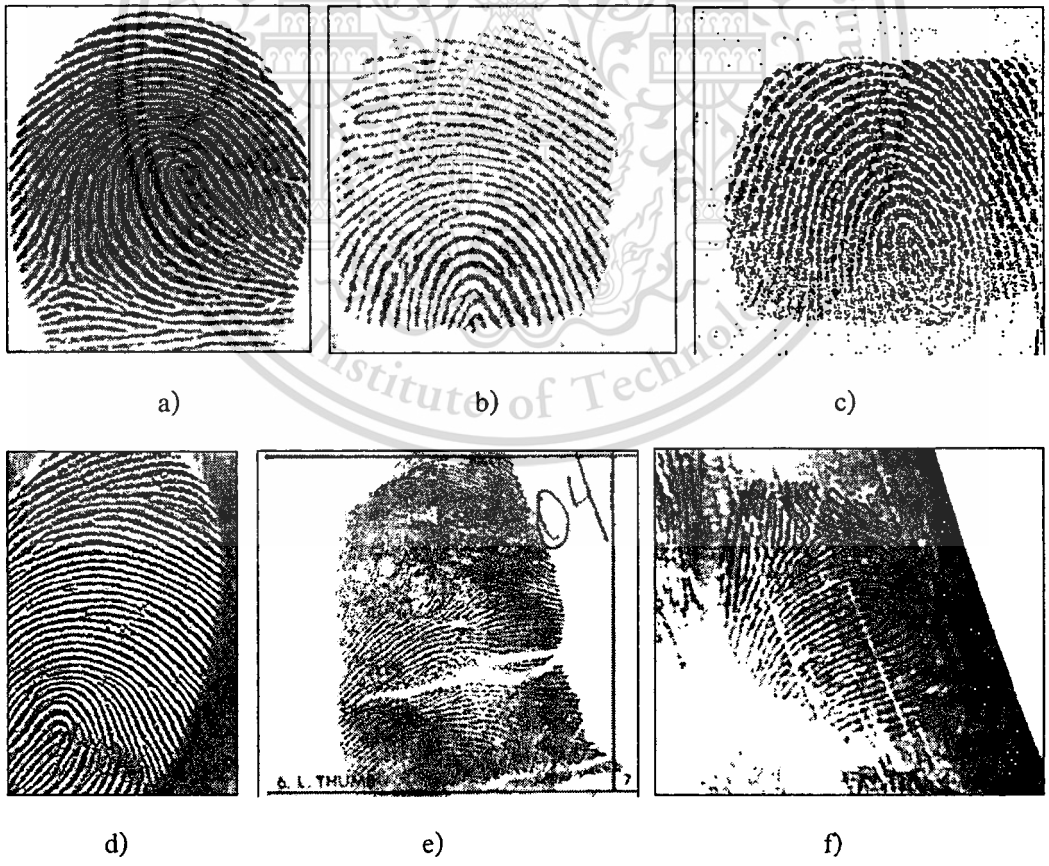


Figure 1.17 Fingerprint images of acquisition.

Figure 1.18 shows some examples of fingerprint sensors embedded in a variety of computer peripherals and other devices. Storing raw fingerprint images may be problematic for large-scale identification systems. In 1995, the size of the FBI fingerprint card archive contained over 200 million items, and archive size was increasing at the rate of 30,000 to 50,000 new cards per day. Although the digitization of fingerprint cards seemed to be the most obvious choice, the resulting digital archive could become extremely large. In fact, each fingerprint card, when digitized at 500 dpi requires about 10 megabytes of storage. A simple multiplication by 200 million yields the massive storage requirement of 2,000 terabytes for the entire archive. An effective compression technique was urgently needed. Unfortunately, neither the well-known lossless methods nor the JPEG methods were found to be satisfactory. A new compression technique (with small acceptable loss), called Wavelet Scalar Quantization (WSQ), became the FBI standard for the compression of 500 dpi fingerprint images. Besides WSQ, a number of other compression techniques (including JPEG2000) have been proposed.

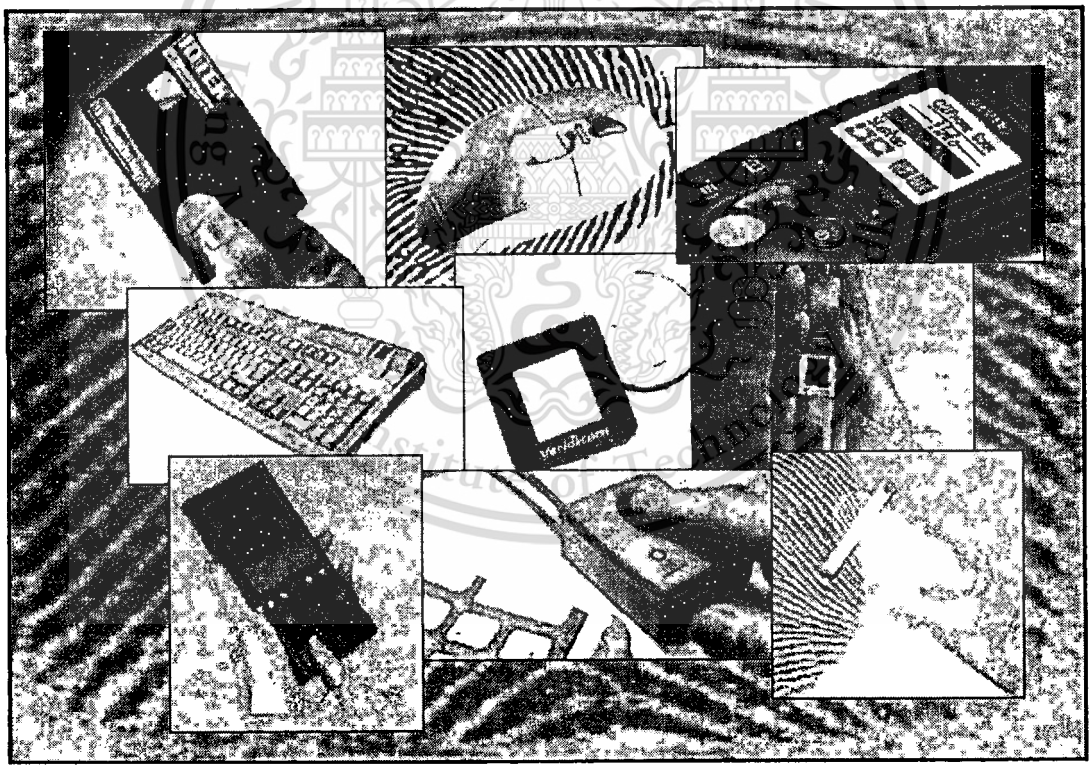


Figure 1.18 Fingerprint sensors can be embedded in a variety of devices for user recognition.

1.10 Fingerprint Representation and Feature Extraction [1]

The representation issue constitutes the essence of fingerprint recognition system design and has far-reaching implications on the matching modules. The pixel intensity values in the fingerprint image are not invariant over time of capture and there is a need to determine salient features of the input fingerprint image that can discriminate between identities as well as remain invariant for a given individual. Thus the problem of representation is to determine a measurement (feature) space in which the fingerprint images belonging to the same finger form a compact cluster (low intra-class variations) and those belonging to different fingers occupy different portions of the space (high inter-class variations).

A good fingerprint representation should have the following two properties: saliency and suitability. Saliency means that a representation should contain distinctive information about the fingerprint. Suitability means that the representation can be easily extracted, stored in a compact fashion, and be useful for matching. A salient representation is not necessarily a suitable representation. In addition, in some biometrics applications, storage space is at a premium. For example, only a few kilobytes of storage are typically available in a smartcard. In such situations, the representation also needs to be compact.

Image-based representations, constituted by pixel intensity information, do not perform well due to factors such as brightness variations, image quality variations, scars, and large global distortions present in fingerprint images. Furthermore, an image-based representation requires a considerable amount of storage. On the other hand, an image-based representation preserves the maximum amount of information and makes fewer assumptions about the application domain. For instance, it is extremely difficult to extract any high level features from a (degenerate) finger devoid of any ridge structure. The fingerprint pattern, when analyzed at different scales, exhibits different types of features.

- Level 1: at the global level, the ridge line flow delineates a pattern similar to one of those shown in Figure 1.19. Singular points, called loop and delta (denoted as squares and triangles, respectively in Figure 1.19), act as control points around which the ridge lines are "wrapped". Singular points and coarse ridge line shape are useful for fingerprint classification (see Chapter 2), but their distinctiveness is not sufficient

for accurate matching. External fingerprint shape, orientation image, and frequency image also belong to the set of features that can be detected at the global level.

- Level 2: at the local level, a total of 150 different local ridge characteristics, called minute details, have been identified. These local ridge characteristics are not evenly distributed. Most of them depend heavily on the impression conditions and quality of fingerprints and are rarely observed in fingerprints. The two most prominent ridge characteristics, called minutiae (see Figure 1.20) are: ridge endings and ridge bifurcations. A ridge ending is defined as the ridge point where a ridge ends abruptly. A ridge bifurcation is defined as the ridge point where a ridge forks or diverges into branch ridges. Minutiae in fingerprints are generally stable and robust to fingerprint impression conditions. Although a minutiae-based representation is characterized by a high saliency, reliable automatic minutiae extraction can be problematic in extremely low-quality fingerprints devoid of any ridge structure.

Figure 1.19 a) left loop, b) right loop, c) whorl, d) arch, and e) tented arch; squares denote loop-type singular points, and triangle delta-type singular points.

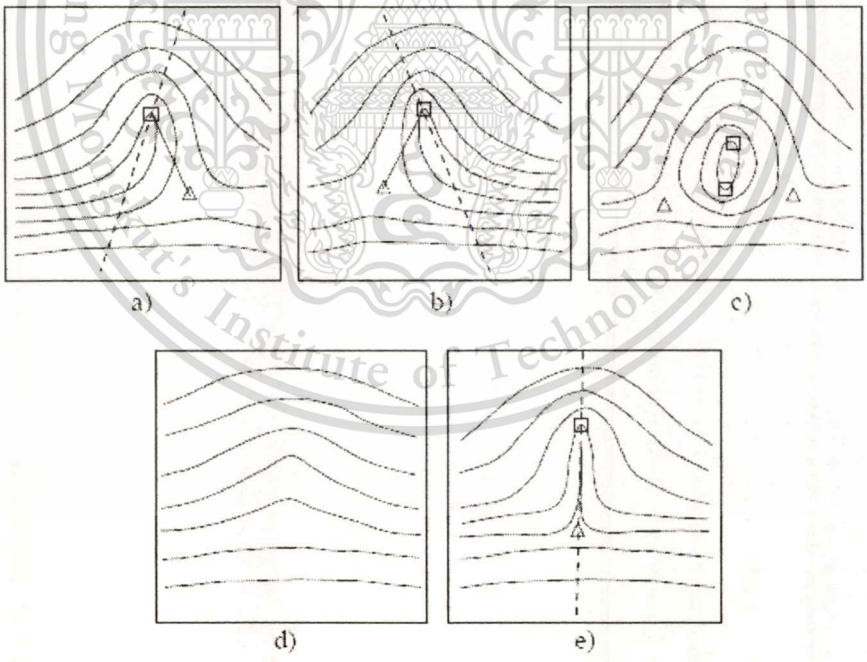


Figure 1.19 Fingerprint patterns as they appear at a coarse level.

- Level 3: at the very-fine level, intra-ridge details can be detected. These include width, shape, curvature, edge contours of ridges as well as other permanent details

such as dots and incipient ridges. One of the most important fine-level details is the finger sweat pores (see Figure 1.20), whose positions and shapes are considered highly distinctive. However, extracting very-fine details including pores is feasible only in high-resolution (e.g., 1,000 dpi) fingerprint images of good quality and therefore this kind of representation is not practical for non-forensic applications.

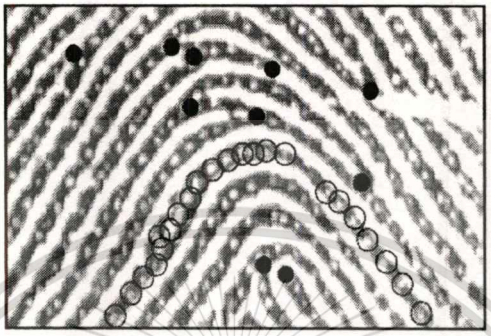


Figure 1.20 Minutiae (black-filled circles) in a portion of fingerprint image; sweat pores (empty circles) on a single ridge line.

1.11 Definition of Fingerprint

Fingerprint is the most important for today's information technology to be able to securely protect information access to the proper authenticated users. Critical information can be lost, stolen or tempered and that can result in lots of opportunities and/or revenues for the company and business. Fingerprint identification is a popular personal identification method for several key reasons: Fingerprints do not change over time; all fingerprints are unique; even identical twins have different sets of fingerprints; fast enrollment and matching of fingerprint; easy to use; low-cost implementation; unique identification is accepted worldwide; and able to store up to ten fingers for each personal enrollment in case of potential injury to the hand.

A fingerprint is the feature pattern of one finger shown in Figure 1.21 a). It is believed with strong evidences that each fingerprint is unique. Each person has his own fingerprints with the permanent uniqueness. So fingerprints have being used for identification and forensic investigation for a long time. A fingerprint is composed of many ridges and valley. These ridges and valley present good similarities in each small local window, like parallelism and average width.

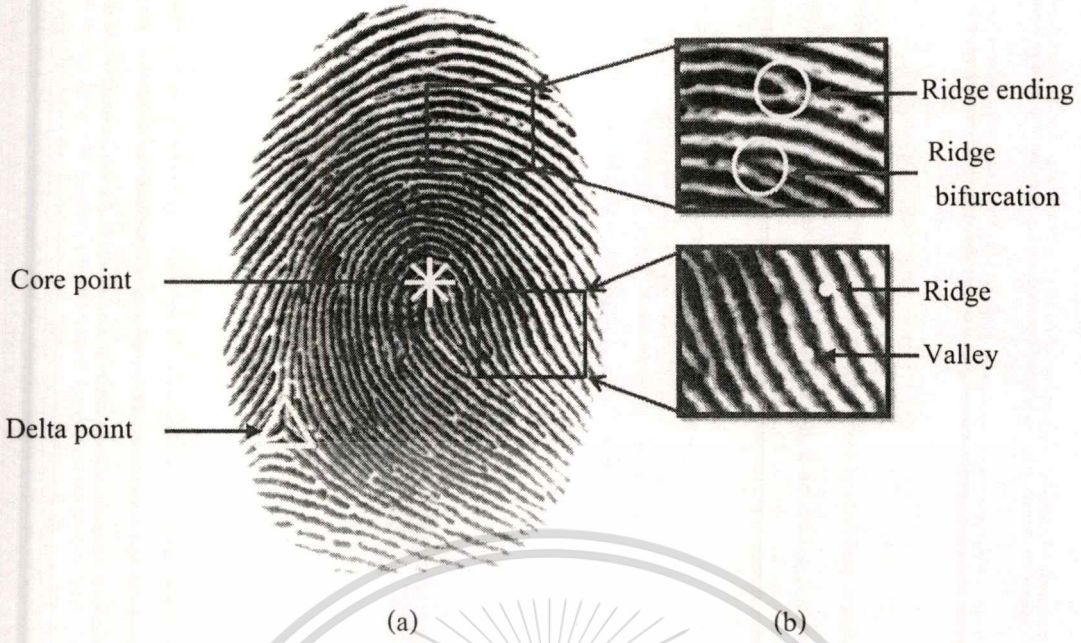


Figure 1.21 Real fingerprint local features (black line).

However, shown by intensive research on fingerprint enhancement, fingerprints are distinguished by their ridges and valley, but by minutia for fingerprint identification, which are some abnormal points on the ridges shown in Figure 1.21 b). Among the variety of minutia types have been reported in fingerprint recognition, two are mostly significant and in heavy usage: one is called termination, which is the immediate ending of a ridge; the other is called bifurcation, which is the point on the ridge from which two branches derive.

1.12 Motivation of This Study

Fingerprint enhancement is the most of important in recognition a person based on his physiological and behavioral characteristics is known as biometrics. Biometrics has gained a lot of importance in recent past due to increase in crime rate and need of more robust and automated security. People from many different backgrounds have contributed for development of biometric system for number of years, for example researchers from fields like sensors, image processing, signal processing, pattern recognition, database management, software development etc. Fingerprints are the most widely used biometric feature for personal identification based on physiological and behavioral characteristics. So that, fingerprint enhancement and fingerprint extracting minutiae is one of the most important steps in automatic fingerprint identification and classification. In practice, there already exists many enhance ridge of fingerprint algorithms.

Almost of them can efficiently enhance ridge of fingerprint, when the fingerprint image is at quite high quality, but when the fingerprint image is of poor quality such as: dry, wet or damp, smudge, scars, blurred and wrinkle the efficient of the algorithm degrades. This study, we propose fingerprint images enhancement with directional filtering to reduce the noise and enhance the definition of ridges, for increasing the contrast between ridges and valley for connecting. The performance of enhanced image is measured for its improvement confirmed by the success of core point identification where Poincare technique is used. The commonly-well-known database FVC-2004 is used for image source. For the case of implementation, the proposed techniques and algorithms are implemented using MATLAB.

1.13 Thesis Outline

After the general introduction of biometric system, various characteristics of application of fingerprint with a background and related works in the area of fingerprint enhancement. In this thesis, we proposed fingerprint image enhancement with directional filtering that involves two types similar of filters: Gabor filter and Second derivative of a Gaussian filter are also applied in the scheme of pyramid technique and directional wavelet transform. The next step gives a brief overview of the most commonly used enhancement algorithms.

Chapter 2 presents fingerprint enhancements, the background information and reviews of brief enhancement algorithm in the literature. The background includes the details of fingerprint matching and classification techniques, the next literature review of fingerprint image enhancement corresponding to our proposed.

Chapter 3 describes the general technique in image processing. In this chapter, a conventional image processing method and literature reviews images enhancement by using filter techniques it common many algorithms present. Further, the filter techniques are taken into consideration for image processing and fingerprint image, which can reduce the noise and improve of the images

Chapter 4 describes fingerprint image enhancement, it is main methodologies for fingerprint image enhancement with directional filtering based on the presents. It is including two types similar of filters Gabor filter and Second derivative of a Gaussian filter are also applied in the scheme of pyramid technique and directional wavelet transform, which can improve the

quality of the ridge fingerprint images for increasing the contrast between ridges and valley for connecting.

Chapter 5 will illustrate the experiment and results. Further, discussion fingerprint enhancement with directional filtering, it is includes Gabor filter and Second derivative of a Gaussian filter are also applied in the scheme of pyramid technique and directional wavelet transform. The performance of enhancement image is measured for its improvement by testing the success of core point identification where Poincare technique is applied.

Chapter 6 presents the main of contribution this thesis, discussion, conclusions and future extension possibility of this work.

1.14 Summary

In this chapter we have introduced the general aspect of biometric; including systems, characteristics. Biometrics are generally divided into two groups; namely physiological and behavioral characteristics. Following, we shifted to fingerprint, another personal unique identifier. The sub-topic includes history, application systems, definition, individuality and extraction. Before moving to the fingerprint enhancement in next chapter, this chapter is concluded with research motivation and the brief on the rest chapters.

Chapter 2

Fingerprint Enhancement

2.1 Needs of Fingerprint Enhancement

Fingerprints are widely used for authentication purpose due to their proved distinctiveness. This uniqueness comes from unique local characteristics ridge bifurcations and ridge endings known as minutiae. Most of the fingerprint matching algorithms use details of these minutiae points to compare two fingerprint images. The performance of these algorithms depends significantly on the quality of fingerprint images. In many cases, fingerprint with numerous discontinuous ridges poor quality such as dry, wet, damped, scars and smudges can cause errors in fingerprint identification process, and also quality of fingerprint images differs due to many factors like imaging conditions, type of sensor is used, acquisition conditions, age, skin characteristics etc. Minutiae from poor quality images contain many pseudo-minutiae. These pseudo-minutiae can affect the performance of a matching algorithm significantly. Therefore, we need an effective fingerprint image enhancement procedure to eliminate these pseudo-minutiae and thus improve the performance of fingerprint matching algorithm.

The main aim an enhancement algorithm is not only to reduce unwanted noise in the image as much as possible and to improve the clarity of ridge structures of fingerprint images in recoverable regions but also to remove the unrecoverable regions. The process is employed in order to improve the clarity of ridge structures of input fingerprint. There are numbers of fingerprint enhancement procedures proposed in the literature. Most of the fingerprint enhancement techniques make the use of filters for enhancement [3, 57-69]. The most widely used technique is based on the use of directional filtering [3, 23, 57-63]. Instead of using filter for the entire image, different filters are used for different areas of the image [64-69]. Filters are designed according to the local ridge orientation and local ridge frequency as they differ in different areas of a fingerprint. An appropriate filter that is tuned to the local ridge frequency and orientation can effectively remove the undesired noise while preserving the true ridge and valley structure.

2.2 Fingerprint Matching [1]

A fingerprint matching algorithm compares two given fingerprints and returns either a degree of similarity (without loss of generality, a score between 0 and 1) or a binary decision (mated/non-mated). Only a few matching algorithms operate directly on grayscale fingerprint images; most of them require that an intermediate fingerprint representation be derived through a feature extraction stage. Without loss of generality, hereafter we denote the representation of the fingerprint acquired during enrollment as the template ($\boldsymbol{T}$) and the representation of the fingerprint to be matched as the input ($\boldsymbol{I}$). In case no feature extraction is performed, the fingerprint representation coincides with the grayscale fingerprint image itself; hence, throughout this chapter, we denote both raw fingerprint images and fingerprint feature vectors (i.e., minutiae) with $\boldsymbol{T}$ and $\boldsymbol{I}$.

The fingerprint feature extraction and matching algorithms are usually quite similar for both fingerprint verification and identification problems. This is because the fingerprint identification problem (i.e., searching for an input fingerprint in a database of N fingerprints) can be implemented as a sequential execution of N one-to-one comparisons (verifications) between pairs of fingerprints. Approaches to fingerprint matching can be coarsely classified into three families.

Correlation-based matching: two fingerprint images are superimposed and the correlation between the corresponding pixels is computed for different alignments (i.e., various displacements and rotations).

Minutiae-based matching: this is the most popular and widely used technique, being the basis of the fingerprint comparison made by fingerprint examiners. Minutiae are extracted from the two fingerprints and stored as sets of points in the two dimensional plane. Minutiae-based matching essentially consists of finding the alignment between the template and the input minutiae feature sets that result in the maximum number of minutiae pairings.

Non-Minutiae feature-based matching: minutiae extraction is difficult in extremely low-quality fingerprint images. While some other features of the fingerprint ridge pattern (i.e., local orientation and frequency, ridge shape, texture information) may be extracted more reliably than minutiae, their distinctiveness as well as persistence is generally lower.

2.2.1 Correlation-based techniques

Let $\boldsymbol{T}$ and $\boldsymbol{I}$ be the two fingerprint images corresponding to the template and the input fingerprint, respectively. Then an intuitive measure of their diversity is the sum of squared differences (*SSD*) between the intensities of the corresponding pixels:

$$SSD(T,I) = \|T-I\|^2 = (T-I)^T(T-I) = \|T\|^2 + \|I\|^2 - 2T^T I \tag{2.1}$$

where the superscript "*T*" denotes the transpose of a vector. If the terms $\|\boldsymbol{T}\|^2$ and $\|\boldsymbol{I}\|^2$ are constant, the diversity between the two images is minimized when the cross-correlation (*CC*) between $\boldsymbol{T}$ and $\boldsymbol{I}$ is maximized:

$$CC(T,I) = T^T I \tag{2.2}$$

Note that the quantity -2 × $CC(\boldsymbol{T},\boldsymbol{I})$ appears as the third term in equation (2.1). The cross-correlation (or simply correlation) is then a measure of the image similarity. Due to the displacement and rotation that unavoidably characterize two impressions of a given finger, their similarity cannot be simply computed by superimposing $\boldsymbol{T}$ and $\boldsymbol{I}$ and applying equation (2.2).

Let $\boldsymbol{I}^{(\Delta_x, \Delta_y, \theta)}$ represent a rotation of the input image $\boldsymbol{I}$ by an angle θ around the origin (usually the image center) and shifted by Δx and Δy pixels in directions x and y, respectively; then the similarity between the two fingerprint images $\boldsymbol{T}$ and $\boldsymbol{I}$ can be measured as

$$S(T,I) = \max_{\Delta x, \Delta y, \theta} CC\left(T, I^{(\Delta x, \Delta y, \theta)}\right) \tag{2.3}$$

A direct application of equation (2.3) rarely leads to acceptable results (see Figure 2.1a) mainly due to the following problems. Figure 2.1 for the best alignment (e.g., that maximize correlation). In the first row, a) the two impressions are very similar and their images correlate well (the residual is very small). In the second row, b) and third row c), due to high distortion and skin condition, respectively, the residuals are high and the global correlation methods fail.

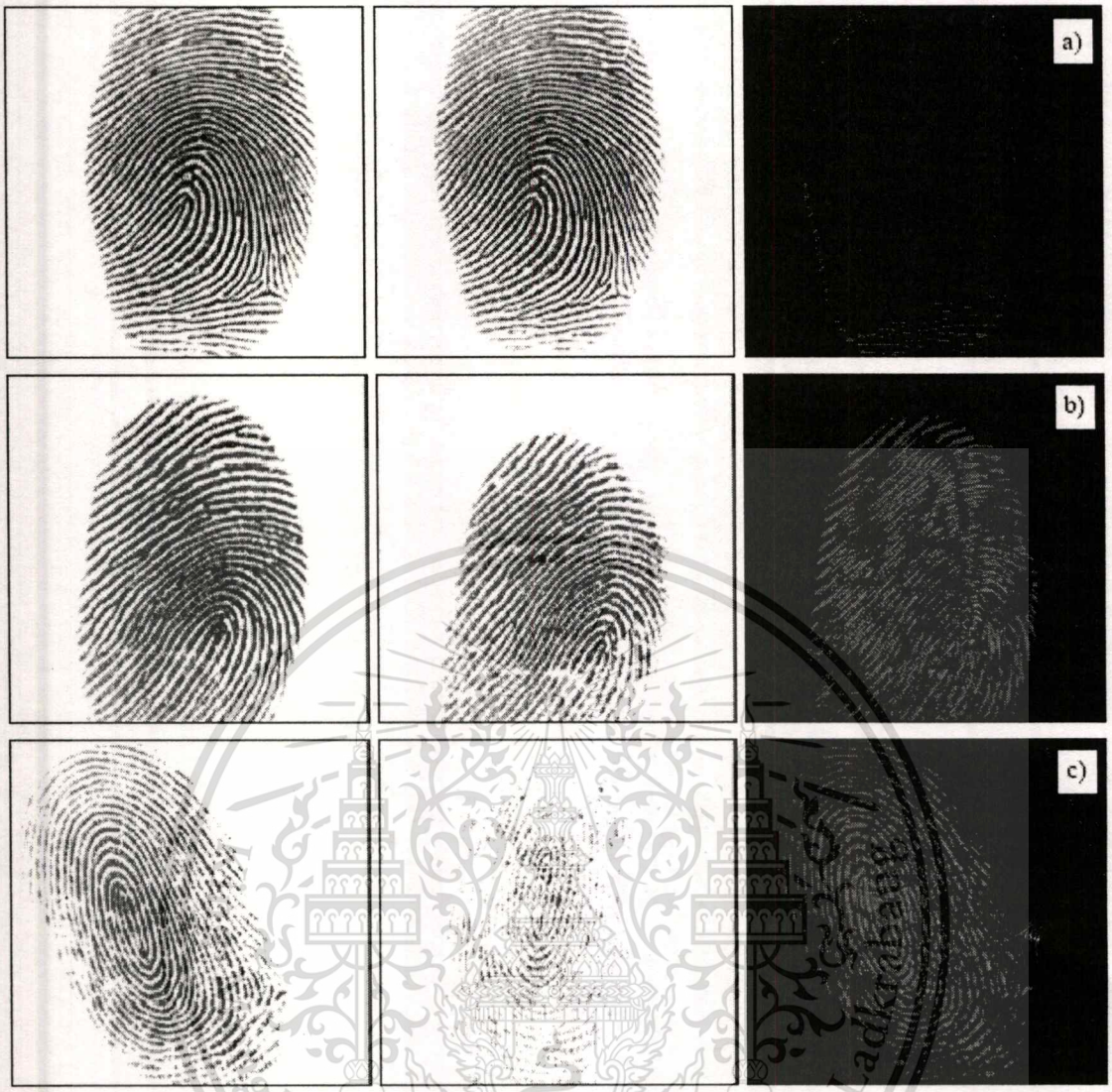


Figure 2.1 Each row shows two impressions of the same finger and the absolute value of their difference (residual).

As to the computational complexity of the correlation technique, smart approaches may be exploited to achieve efficient implementations.

- The correlation theorem [6] states that computing the correlation in the spatial domain (operator $\otimes$) is equivalent to performing a point-wise multiplication in the Fourier domain; in particular,

$$T \otimes I = F^{-1}\left(F^{*}(T) \times F(T)\right) \tag{2.4}$$

where $F(\ )$ is the Fourier transform of an image, $F^{-1}(\ )$ is the inverse Fourier transform, "*" denotes the complex conjugate, and "×" denotes the point-by-point

multiplication of two vectors. The result of Equation (2.4) is a correlation image whose value at the pixel $[x,y]$ denotes the correlation between $\boldsymbol{T}$ and $\boldsymbol{I}$ when the displacement is $\Delta x = x$ and $\Delta y = y$. However the output of Equation (2.4) is dependent on the image energy and the correlation peak (corresponding to the optimal registration) can be small. The Symmetric Phase Only Filter (SPOF) often provides better results (equation (2.5)):

$$T \otimes_{SPOF} I = F^{-1}\left(\frac{F^{*}(T)}{|F(T)|} \times \frac{F(I)}{|F(I)|}\right) \tag{2.5}$$

To reduce the effect of noise [7] suggests restricting the SPOF domain to the frequency range characterizing a fingerprint image: this can be simply dealt with through band-pass filtering in the Fourier space. Equations (2.4) and (2.5) do not take into account rotation, which has to be dealt with separately; in any case, the computational saving is very high when correlation is performed globally [8] and considerable when it is performed locally by using medium-size regions.

- Computing the maximum correlation need not necessarily be done in a sequential, exhaustive manner; multi-resolution approaches, space-searching techniques (e.g., gradient descent), and other heuristics can be adopted to reduce the number of evaluations. For example, [9] propose to coarsely pre-align the two fingerprints based on their orientation images.
- The Fourier-Mellin transform [10] may be used instead of the Fourier transform to achieve rotation invariance in addition to translation invariance; on the other hand, some additional steps (such as the log-polar coordinate transformation) have to be performed, that can reduce the accuracy of this solution. [11] Method computes the Fourier-Mellin descriptors locally and uses SPOF to determine the similarity between any two image portions.
- The approach proposed by [12] partitions both $\boldsymbol{T}$ and $\boldsymbol{I}$ into local regions and computes the maximum correlation (in the Fourier domain) between any pair of regions. This method suffers from "border effects" because of the partial overlapping between the different blocks, but can considerably speed up the whole matching process.

2.2.2 Minutiae-based methods

A minutia matching is certainly the most well-known and most widely used method for fingerprint matching, thanks to its strict analogy with the way forensic experts compare fingerprints and its acceptance as a proof of identity in the courts of law in almost all countries around the world.

1) Problem formulation

Let ***T*** and ***I*** be the representation of the template and input fingerprint, respectively. Unlike in correlation-based techniques, where the fingerprint representation coincides with the fingerprint image, here the representation is a feature vector (of variable length) whose elements are the fingerprint minutiae. Each minutia may be described by a number of attributes, including its location in the fingerprint image, orientation, type (e.g., ridge ending or ridge bifurcation), a weight based on the quality of the fingerprint image in the neighborhood of the minutia, and so on. Most common minutiae matching algorithms consider each minutia as a triplet $m = \{x,y,\theta\}$ that indicates the x,y minutia location coordinates and the minutia angle θ:

$$T = \{m_1, m_2, \cdots, m_m\}, \quad m_i = \{x_i, y_i, \theta_i\}, \quad i = 1 \cdots m$$

$$I = \{m'_1, m'_2, \cdots, m'_n\}, \quad m'_j = \{x'_j, y'_j, \theta'_j\}, \quad j = 1 \cdots n,$$

where m and n denote the number of minutiae in ***T*** and ***I***, respectively.

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

$$sd(m'_j, m_i) = \sqrt{(x'_j - x_i)^2 + (y'_j - y_i)^2} \leq r_0 \tag{2.6}$$

$$dd(m'_j, m_i) = \min\left(\left|\theta'_j - \theta_i\right|, 360° - \left|\theta'_j - \theta_i\right|\right) \leq \theta_0 \tag{2.7}$$

Equation (2.7) takes the minimum of $\left|\theta'_j - \theta_i\right|$ and $360° - \left|\theta'_j - \theta_i\right|$ because of the circularity of angles (the difference between angles of 2° and 358° is only 4°). The *tolerance boxes* (or hyper-spheres) defined by r_0 and θ_0 are necessary to compensate for the unavoidable errors made by feature extraction algorithms and to account for the small plastic distortions that cause the minutiae positions to change.

Aligning the two fingerprints is a mandatory step in order to maximize the number of matching minutiae. Correctly aligning two fingerprints certainly requires displacement (in x and y) and rotation (θ) to be recovered and likely involves compensating for other geometrical transformations:

- Scale has to be considered when the resolution of the two fingerprints may vary (e.g., the two fingerprint images have been taken by scanners operating at different resolutions).
- Other distortion-tolerant geometrical transformations could be useful to match minutiae in case one or both of the fingerprints is affected by severe distortions.

In any case, tolerating more geometric transformations beyond translation and rotation results in additional degrees of freedom to the minutiae matcher: when a matcher is designed, this issue needs to be carefully evaluated, as each degree of freedom results in a huge number of new possible alignments which significantly increases the chance of incorrectly matching two fingerprints from different fingers:

Let $map(\ )$ be the function that maps a minutia m'_j (from **I**) into m''_j according to a given geometrical transformation; for example, by considering a displacement of $[\Delta x, \Delta y]$ and a counterclockwise rotation θ around the origin1:

$$map_{\Delta x,\Delta y,\theta}\left(m'_j=\left\{x'_j,y'_j,\theta'_j\right\}\right)=m''_j=\left\{x''_j,y''_j,\theta'_j+\theta\right\}\ , \qquad \text{where,}$$

$$\begin{bmatrix}x''_j\\ y''_j\end{bmatrix}=\begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix}\begin{bmatrix}x'_j\\ y'_j\end{bmatrix}+\begin{bmatrix}\Delta x\\ \Delta y\end{bmatrix}$$

Let $mm(\ )$ be an indicator function that returns 1 in the case where the minutiae m''_j and m_i match according to Equations (2.6) and (2.7):

$$mm\left(m''_j,m_i\right)=\begin{cases}1 & sd\left(m''_j,m_i\right)\le r_0 \quad and \quad dd\left(m''_j,m_i\right)\le\theta_0\\ 0 & otherwise.\end{cases}$$

Then, the matching problem can be formulated as

$$\max_{\Delta x,\Delta y,\theta,P} imize \sum mm\left(map_{\Delta x,\Delta y,\theta}\left(m'_{P(i)}\right),m_i\right) \qquad (2.8)$$

where $P(i)$ is an unknown function that determines the *pairing* between $\boldsymbol{I}$ and $\boldsymbol{T}$ minutiae; in particular, each minutia has either exactly one mate in the other fingerprint. Figure 2.2 shows an example of minutiae pairing given a fingerprint alignment. To achieve the optimum pairing (according to equation (2.8)), a slightly more complicated scheme should be adopted: in fact, in the case when a minutia of $\boldsymbol{I}$ falls within the tolerance hyper-sphere of more than one minutia of $\boldsymbol{T}$, the optimum assignment is that which maximizes the number of mates (refer to Figure 2.3 for a simple example). Figure 2.2 minutiae of T are denoted by **o**, whereas I minutiae are denoted by ×. Note that $\boldsymbol{I}$ minutiae are referred to as m'', because what is shown in the figure is their mapping into $\boldsymbol{T}$ coordinates. Pairing is performed according to the minimum distance. The dashed circles indicate the maximum spatial distance. The gray circles denote successfully mated minutiae; minutia m_1 of T and minutia m_3'' of $\boldsymbol{I}$ have no mates, minutiae m_3 and m_6'' cannot be mated due to their large direction difference [13].

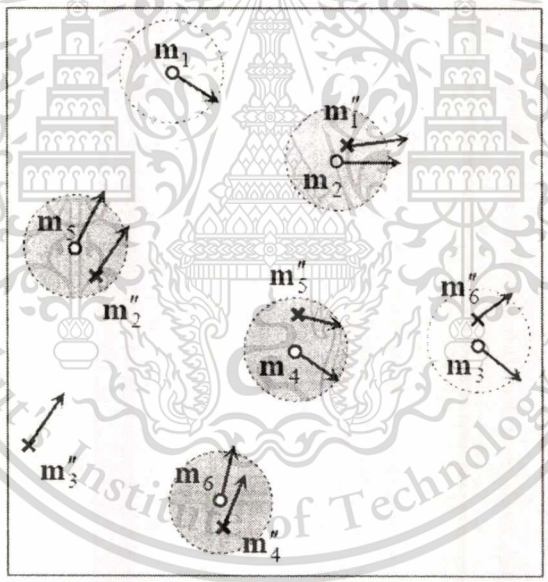


Figure 2.2 Minutiae of I mapped into $\boldsymbol{T}$ coordinates for a given alignment.

As shown in Figure 2.3 if m_1 was mated with m_2'' (the closest minutia), m_2 would remain unmated; however, pairing m_1 with m_1'', allows m_2 to be mated with m_2'', thus maximizing equation (2.8).

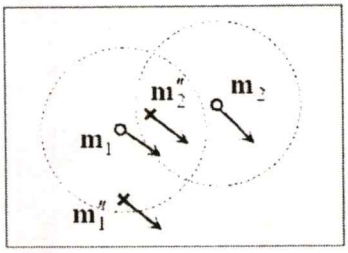


Figure 2.3 In this example of mated minutia.

2) Similarity score

Unlike in manual matching performed by forensic experts where the number of matching minutiae is itself the main output of the comparison, automatic matching systems must convert this number into a similarity score. This is often performed by simply normalizing the number of matching minutiae (here denoted by k) by the average number $(m + n)/2$ of minutiae in $\boldsymbol{T}$ and $\boldsymbol{I}$:

$$score = \frac{k}{(n+m)/2} \tag{2.9}$$

However, further information can be exploited, especially in case of noisy images and limited overlap between $\boldsymbol{T}$ and $\boldsymbol{I}$, to compute a more reliable score; in fact:

- Minutiae quality can be used to weight differently reliable and unreliable minutiae pairs: the contribution from a pair of reliable minutiae should be higher than that from a pair where at least one the two minutiae are of low quality [14]. The quality of a minutia (and of a minutia pair) can be defined according to the fingerprint quality in the region where the minutia lies and/or by keeping into account other local information.
- The normalization in equation (2.9) tends to excessively penalize fingerprint pairs with partial overlap; a more effective normalization considers the number or minutiae belonging to the intersection of the two fingerprints after the optimal alignment have been determined [15].

3) Point pattern matching

The minutiae matching problem can also be viewed as a point pattern matching problem. Because of its central role in many pattern recognition and computer vision tasks (i.e., object matching, remote sensing, camera calibration, motion estimation), point pattern matching has

been extensively studied yielding families of approaches known as algebraic geometry, Hough transform, relaxation, Operations Research solutions, energy-minimization, and so on.

- Algebraic geometry: several methods have been proposed in the literature for different versions of the problem: $n = m$ or $n \leq m$, exact or inexact point matching; see [16] for a survey proposed an algorithm to perform an inexact partial point pattern matching with $O(m^2 \times n^2 \times log\ m)$ time complexity. However, this algorithm makes some simplifying assumptions that are not always fulfilled by minutiae points; in fact, the algorithm requires that: (*i*) all the points in ***I*** have a mate in ***T***, even if some points in ***T*** can have no mate in ***I***, and (*ii*) the tolerance boxes around the points do not intersect each other or, equivalently, that the points in ***T*** are not too close to each other.
- Hough transform: the generalized Hough transform-based approach [17] converts point pattern matching to the problem of detecting peaks in the Hough space of transformation parameters. It discretizes the parameter space and accumulates evidence in the discretized space by deriving transformation parameters that relate two sets of points using a substructure of the feature matching technique.
- Relaxation: the relaxation approach [18] iteratively adjusts the confidence level of each corresponding pair of points based on its consistency with other pairs until a certain criterion is satisfied. At each iteration r, the method computes $m \times n$ probabilities p_{ij} (probability that point i corresponds to point j):

$$p_{ij}^{(r+1)} = \frac{1}{m}\sum_{h=1}^{m}\left[\max_{k=1\cdots n}\left\{c(i,j;h,k)\,p_{ij}^{(r)}\right\}\right], \quad i=1\cdots m, \quad j=1\cdots n, \tag{2.10}$$

where $c(i, j; h, k)$ is a compatibility measure between the pairing (i, j) and (h, k), which can be defined according to the consistency of the alignments necessary to map point j into i and point k into h. Equation (10) increases the probability of those pairs that receive substantial support by other pairs, and decreases the probability of the remaining ones. At convergence, each point i may be associated with the point j such that $p_{ij} = \max_s \{p_{is}\}$, where ***s*** is any other point in the set.

- Operations Research solutions: tree-pruning approaches attempt to find the correspondence between the two point sets by searching over a tree of possible matches while employing different tree-pruning methods (e.g., branch and bound) to

reduce the search space [19]. To prune the tree of possible matches efficiently, this approach tends to impose a number of requirements on the input point sets, such as an equal number of points ($n = m$) and no outliers (points without correspondence). These requirements are difficult to satisfy in practice, especially in fingerprint minutiae matching. A minutiae matching algorithm based on minimum spanning tree matching was proposed by [20].

- Energy minimization: these methods define a function that associates an *energy* or *fitness* with each solution of the problem. Optimal solutions are then derived by minimizing the energy function (or maximizing fitness) by using a stochastic algorithm such as the Genetic algorithm [21] provided specific Genetic algorithm implementations for global minutiae matching. In general, the methods belonging to this category tend to be slow and are unsuitable for real-time minutiae matching.

4) Minutiae matching with pre-alignment

Embedding fingerprint alignment into the minutiae matching stage (as the methods presented in the previous section do), certainly leads to the design of robust algorithms, which are often able to operate with noisy and incomplete data. On the other hand, the computational complexity of such methods does not provide a high matching throughput (i.e., 10,000 or more matches per second), as required by AFIS or civil systems.

Storing pre-aligned templates in the database and pre-aligning the input fingerprint before the minutiae matching can be a valid solution to speed up the 1:*N* identification. In theory, if a perfect pre-alignment could be achieved, the minutiae matching could be reduced to a simple pairing. Two main approaches for pre-alignment have been investigated.

- Absolute pre-alignment: each fingerprint template is pre-aligned, independently of the others, before storing it in the database. Matching an input fingerprint ***I*** with a set of templates requires ***I*** to be independently registered just once, and the resulting aligned representation to be matched with all the templates. The most common absolute pre-alignment technique translates the fingerprint according to the position of the core point. Unfortunately, reliable detection of the core is very difficult in noisy images and in arch type patterns, and a registration error at this level is likely to result in a matching error. Absolute pre-alignment with respect to rotation is even more critical; some authors proposed using the shape of the external fingerprint

silhouette (if available), the orientation of the core delta segment (if a delta exists), the average orientation in some regions around the core or the orientations of the singularities [22].

- Relative pre-alignment: the input fingerprint ***I*** have to be pre-aligned with respect to each template ***T*** in the database; 1: *N* identification requires *N* independent pre-alignments. Relative pre-alignment may determine a significant speed up with respect to the algorithms that do not perform any pre-alignment, but cannot compete in terms of efficiency with absolute pre-alignment. However, relative pre-alignment is in general more effective (in terms of accuracy) than absolute pre-alignment, because the features of the template ***T*** may be used to drive the registration process.

Figure 2.4 a) input minutiae set; b) template minutiae set; c) alignment result based on the minutiae marked with green circles; d) matching result where template minutiae and their correspondences are connected by green lines [23] .

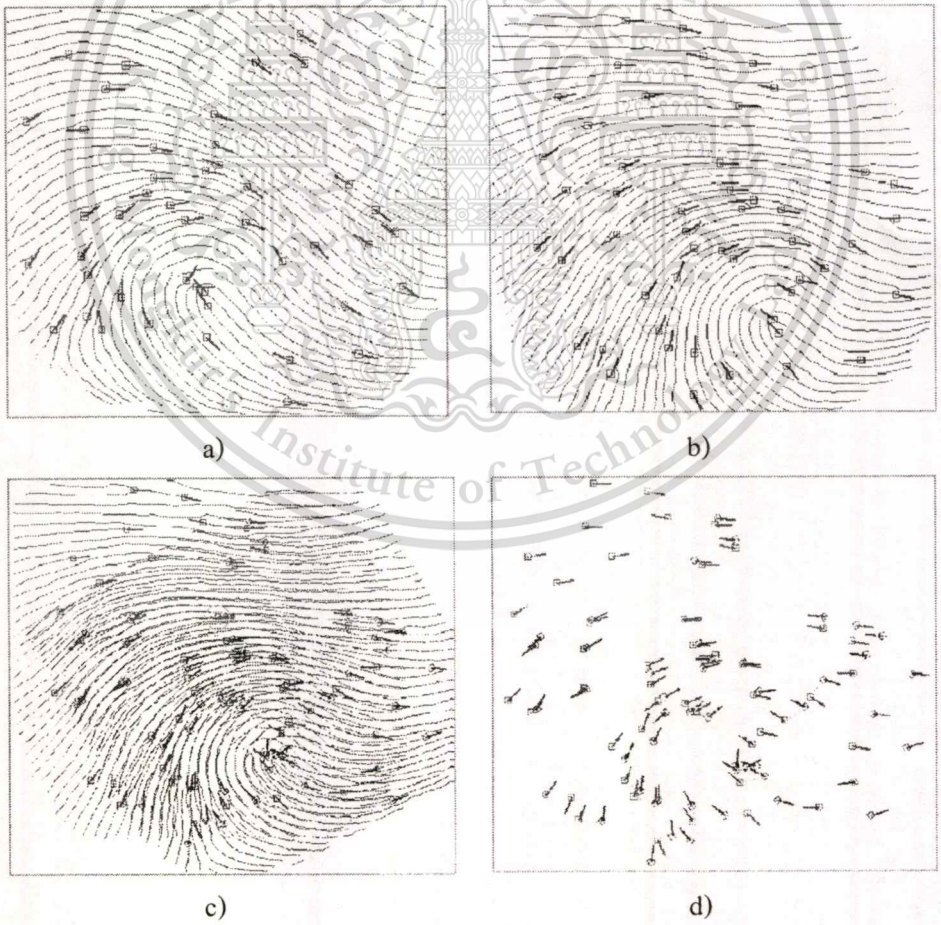


Figure 2.4 Results of applying the matching algorithm to an input minutiae set and a template.

5) Avoiding alignment

Fingerprint alignment is certainly a critical and time-consuming step. To overcome problems involved in alignment, and to better cope with local distortions, some authors perform minutiae matching locally. A few other attempts have been proposed that try to globally match minutiae without requiring explicit recovery of the parameters of the transformation. An introduced an intrinsic coordinates system (ICS) whose axes run along hypothetical lines defined by the local orientation of the fingerprint pattern [24]. First, the fingerprint is partitioned in regular regions (i.e., regions that do not contain singular points). In each regular region, the ICS is defined by the orientation field. When using intrinsic coordinates instead of pixel coordinates, minutiae are defined with respect to their position in the orientation field (Figure 2.5). Translation, displacement, and distortion move minutiae with the orientation field they are immersed in and therefore do not change their intrinsic coordinates. On the other hand, some practical problems such as reliably partitioning the fingerprint in regular regions and unambiguously defining intrinsic coordinate axes in low-quality fingerprints still remain to be solved. Figure 2.5 a) the fingerprint is partitioned into four regular regions; b) each region is spanned by two axes, one parallel to the ridge orientation, and the other perpendicular to the ridges; c) iso-coordinates in the ICS spaces; d) intrinsic coordinates of a given minutia. Reprinted with permission from [24]

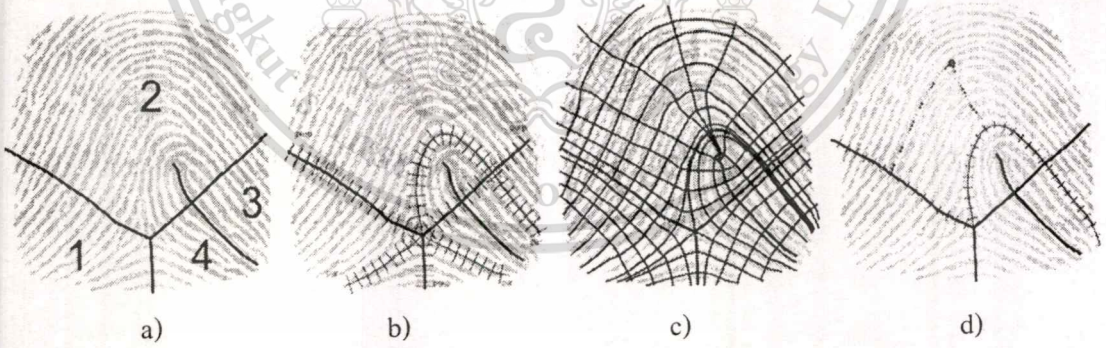


Figure 2.5 Intrinsic Coordinate System (ICS).

2.2.3 Non-minutiae feature-based matching techniques

Three main reasons induce designers of fingerprint recognition techniques to search for additional fingerprint distinguishing features, beyond minutiae:

- Additional features may be used in conjunction with minutiae (and not as an alternative) to increase system accuracy and robustness. It is worth noting that several non-minutiae feature based techniques use minutiae for pre-alignment or to define anchor points.
- Reliably extracting minutiae from extremely poor quality fingerprints is difficult. Although minutiae may carry most of the fingerprint discriminatory information, they do not always constitute the best tradeoff between accuracy and robustness for the poor quality fingerprints.
- Non-minutiae-based methods may perform better than minutiae-based methods when the area of fingerprint sensor is small. In fingerprints with small area, only 4–5 minutiae may exist and in that case minutiae-based algorithm do not behave satisfactorily.

1) Global and local texture information

Global and local texture information sources are important alternatives to minutiae, and texture- based fingerprint matching is an active area of research. Image texture is defined by spatial repetition of basic elements, and is characterized by properties such as scale, orientation, frequency, symmetry, isotropy, and so on. Fingerprint ridge lines are mainly described by smooth ridge orientation and frequency, except in singular regions. These singular regions are discontinuities in a basically regular pattern and include the loop(s) and the delta(s) at a coarse resolution and the minutiae points at a high resolution.

We know that most of the local texture information is contained in the orientation and frequency images. Several methods have been proposed where a similarity score is derived from the correlation between the aligned orientation images of the two fingerprints: [25, 26]. The alignment can be based on the orientation image alone or delegated to a further minutiae matching stage. Also [27] used orientation image in their hybrid matching technique, but unlike other researchers they used a model based approximation of the orientation image. Their experimental results show that, not only this allows to reduce the template size (in fact only the model parameters have to be stored), but also improve accuracy due to the robustness of the model to local perturbation of the orientation image. In [28] used frequency image correlation in conjunction with minutiae matching to produce a final score. Since the frequency images directly computed from the fingerprints may lack in robustness against noise and distortion, a polynomial

model is proposed to approximate the coarse frequency map, thus making the resulting feature more robust.

The most popular technique to match fingerprints based on texture information remains the FingerCode approach by [29]. The fingerprint area of interest is tessellated with respect to the core point (see Figure 2.6 [29]). A feature vector is composed of an ordered enumeration of the features extracted from the local information contained in each sector specified by the tessellation. Thus the feature elements capture the local texture information and the ordered enumeration of the tessellation captures the global relationship among the local contributions. The local texture information in each sector is decomposed into separate channels by using a Gabor filterbank; in fact, the Gabor filterbank is a well-known technique for capturing useful texture information in specific bandpass channels as well as decomposing this information into biorthogonal components in terms of spatial frequencies. In their experimentation, obtained good results by tessellating the area of interest into 80 cells (five bands and 16 sectors), and by using a bank of eight Gabor filters (eight orientations, 1 scale = 1/10 for 500 dpi fingerprint images). Therefore, each fingerprint is represented by an $80 \times 8 = 640$ fixed-size feature vector, called the *FingerCode*. The generic element V_{ij} of the vector ($i = 1 \ldots 80$ is the cell index, $j = 1 \ldots 8$ is the filter index) denotes the energy revealed by the filter j in cell i, and is computed as the average absolute deviation (AAD) from the mean of the responses of the filter j over all the pixels of the cell i:

$$V_{ij} = \frac{1}{n_i}\left[\sum_{C_i}\left|g\left(x, y : \theta_j, 1/10\right) - \overline{g_i}\right|\right] \tag{2.11}$$

where C_i is the ith cell of the tessellation, n_i is the number of pixels in C_i, the Gabor filter expression $g(\,)$ is defined by Equation (..) in chapter 4 and $\overline{g_i}$ is the mean value of g over the cell C_i. Matching two fingerprints is then translated into matching their respective FingerCodes, which is simply performed by computing the Euclidean distance between two FingerCodes.

2) Geometrical attributes and spatial relationship of the ridge lines

The use of spatial relationship of ridges forms the basis of the earlier methods proposed. In the former, tree grammars are introduced to classify ridge line patterns after they are binarized and thinned. In the latter, incremental graph matching was carried out to compare a set of ridges arranged in graph structures (ridges are the graph nodes and arches are defined according to ridge adjacency and visibility). More recently other researchers focused on techniques relying (at least

partially) on ridge matching. The ridges representation is usually obtained by sampling points (at fixed intervals) along each thinned ridge.

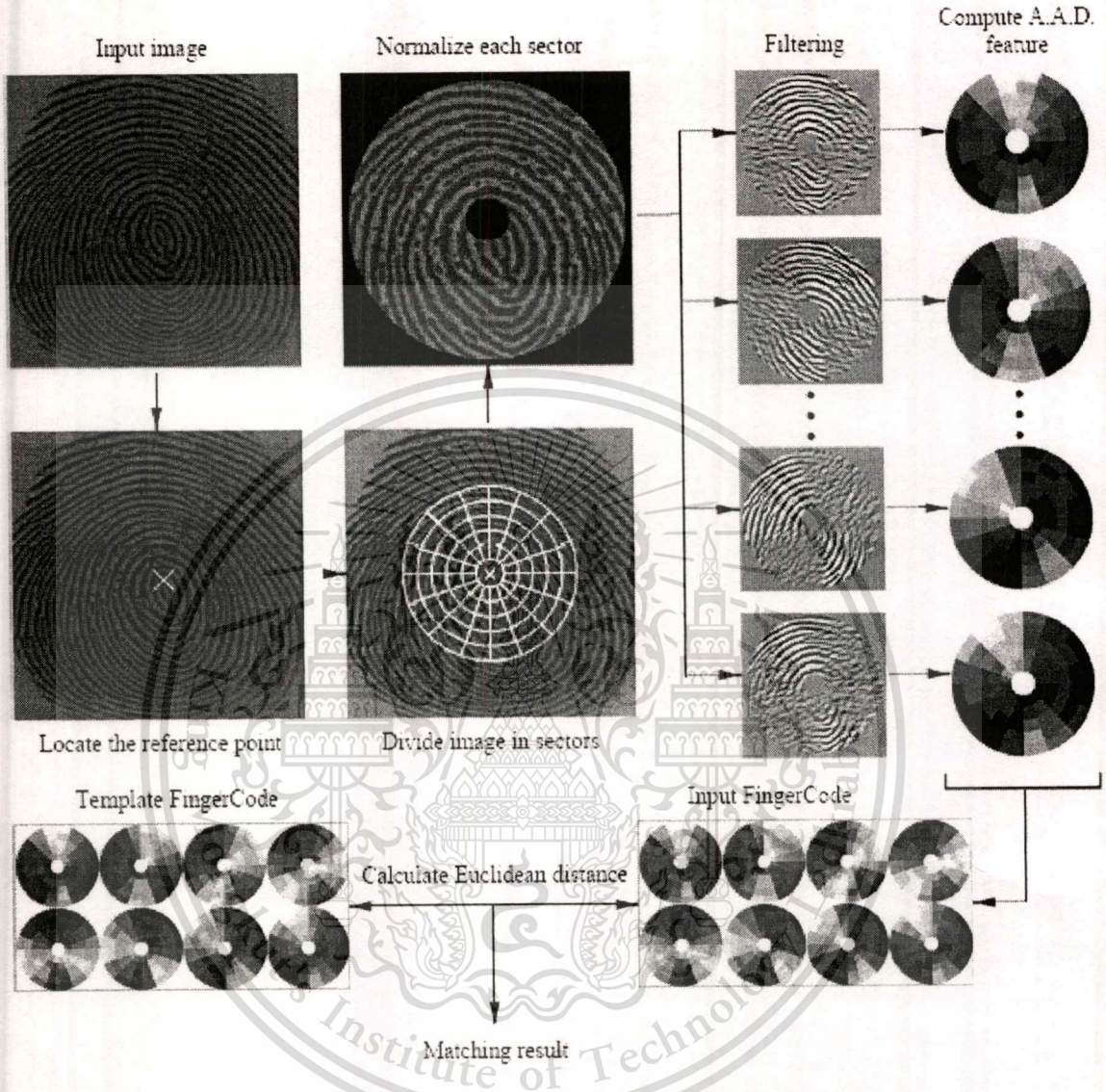


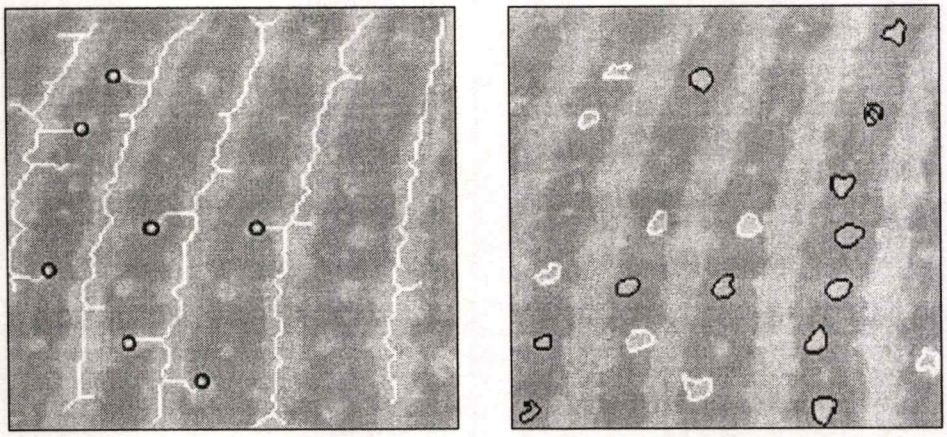
Figure 2.6 System diagram of FingerCode approach.

In [30] ridge information is associated with the minutiae they originate from; once the minutiae are paired with a local structure based algorithm, the computation of the final score is consolidated by taking into account the spatial matching of the ridge points; experimental results show that the contribution by the ridge sampling is significant. [31, 32] explicitly exploit ridge relationships: for each sampled point they annotate the labels of the two neighboring ridges. More details are given below:

- In [31] ridge-matching is performed by comparing neighboring information for all the pre-matched pairs of ridges: two ridges (one from **T** and one from **I**) are pre-matched if they have similar length and curvature. The ridge matching implementation outperforms a minutiae matching implementation of the same authors over the four FVC2000 databases.
- In [32] the ridge and minutiae matching is performed incrementally, starting from a local structure based alignment.
- In [33] encode the geometric relationships among points sampled over ridges based on RCS (Ridge Coordinate System); an RCS is defined for every ridge *R* by using a minutia to define the origin and the direction of the reference axes and the ridge count as a metric for measuring distances. The RCS coordinates of the points of all the ridges (except *R*) form the feature vector associated to *R*. These feature vectors, which are invariant under translation and rotation, are used for alignment and matching through a greedy algorithm.

3) **Level 3 features**

Arrangement of sweat pores along fingerprint ridges is undoubtedly highly discriminate but, as pointed out by several researchers, reliable detection of sweat pores requires high-resolution scanners and robust extraction algorithms. In [34] proposed a skeletonization-based pore extraction and matching algorithm. Specifically, the locations of all end points (with at most one neighbor) and branch points (with exactly three neighbors) in the skeleton image are extracted and each end point is used as a starting location for tracking the skeleton. The tracking algorithm advances one element at a time until one of the following stopping criteria is encountered: (1) another end point is detected, (2) a branch point is detected, and (3) the path length exceeds a maximum allowed value. Condition (1) implies that the tracked segment is a closed pore, while Condition (2) implies an open pore. Finally, skeleton artifacts resulting from scars and wrinkles are corrected and pores from reconnected skeletons are removed. An example of pore extraction is shown in Figure 2.7 [34], a) detection of open pores from the skeleton; b) extraction of open pores (white) and closed pores (black).



a) b)

Figure 2.7 Examples of skeletonization-based pore extraction and matching algorithm.

The pore matching is performed (in a human-assisted or semi-automatic mode) by: *(i)* subdividing the images in small regions and retaining only discriminate regions; *(ii)* pre-aligning the regions through intensity correlation; *(iii)* counting the numbers of spatially coincident pores for each pair of regions. In [35] studied the distinctiveness of minutiae and pore features in fragmentary fingerprint comparison by using images acquired at about 2,000 dpi. For this purpose they also implemented a skeletonization-based pore detection algorithm with more anatomical constraints with respect to [34] algorithm. [35] Concluded that the use of pores can offer at least a comparable recognition potential from a small area fingerprint fragment, as minutiae features offer for fragments of larger area.

In [36] conducted a systematic study to determine how much performance gain one can achieve by introducing Level 3 features in AFIS. They noted that skeletonization is effective for pore extraction only when the image quality is very good and the image resolution is very high. Hence, unlike in previous studies, Level 3 features, including pores and ridge contours, were extracted using Gabor filters and wavelet transform: see Figure 2.8 for some details of a pore extraction process. Level 1 (orientation images), Level 2 (minutiae) and Level 3 (pores and ridge contours) features are then hierarchically matched, leading to early rejection in case of very low similarity. Pores and ridge contours are locally matched (in the neighborhoods of the already paired minutiae) using the Iterative Closest Point (ICP) algorithm. [36] Evaluated their approach over a database acquired with a 1,000 dpi commercial scanner and showed that Level 3 features carry significant discriminatory information. There is a relative reduction of 20% in the EER of the matching system when Level 3 features were employed in combination with Level 1 and 2

features. The performance gain is consistently observed across various quality fingerprint images. Figure 2.8 in [36] a) a partial fingerprint image at 1,000 ppi; b) enhancement of the image shown in a) using a Gabor filter-based contextual technique. c) A linear combination of a) and b); d) wavelet band-pass filtering of the image in a) that exhibits small dark blob in correspondence of the pores; e) a linear combination of d) and b); f) extracted pores (white circles) after thresholding the image in e).

Figure 2.8 Pores extraction.

2.3 Fingerprint Classification [1]

The identification of a person requires a comparison of her fingerprint with all the fingerprints in a database. This database may be very large (e.g., several million fingerprints) in many forensic and civilian applications. In such cases, the identification typically has an unacceptably long response time. The identification process can be speeded up by reducing the number of comparisons that are required to be performed. Sometimes, information about sex, race, age, and other data related to the individual are available and the portion of the database to be searched can be significantly reduced; however, this information is not always accessible (e.g.,

criminal identification based on latent fingerprints) and, in the general case, information intrinsic to the biometric samples has to be used for an efficient retrieval. A common strategy to speed up the search is to divide the fingerprint database into a number of bins (based on some predefined classes). A fingerprint to be identified is then required to be compared only to the fingerprints in a single bin of the database based on its class.

Fingerprint classification refers to the problem of assigning a fingerprint to a class in a consistent and reliable way. Although fingerprint matching is usually performed according to local features (e.g., minutiae), fingerprint classification is generally based on global features, such as global ridge structure and singularities. In fact, most of the classification schemes currently used by law enforcement agencies worldwide are variants of the Galton-Henry classification scheme. Figure 2.9 shows the five most common classes of the Galton-Henry classification scheme (arch, tented arch, left loop, right loop, and whorl):

- An arch fingerprint has ridges that enter from one side, rise to a small bump, and go out the opposite side from which they entered. Arches do not have loops or deltas.
- A tented arch fingerprint is similar to the (plain) arch, except that at least one ridge exhibits a high curvature and one loop and one delta are present.
- A loop fingerprint has one or more ridges that enter from one side, curve back, and go out the same side they entered. A loop and a delta singularities are present; the delta is assumed to be south of the loop. Loops can be further subdivided: loops that have ridges that enter and leave from the left side are called left loops and loops that have ridges that enter and leave from the right side are called right loops.
- A whorl fingerprint contains at least one ridge that makes a complete 360° path around the center of the fingerprint. Two loops (or a whorl) and two deltas can be found in whorl fingerprints. The whorl class is quite complex and in some classification schemes, it is further divided into two categories: twin loop (or double loop) and plain whorl (see Figure 2.9) two whorl fingerprints are shown (a plain whorl and a twin loop, respectively).

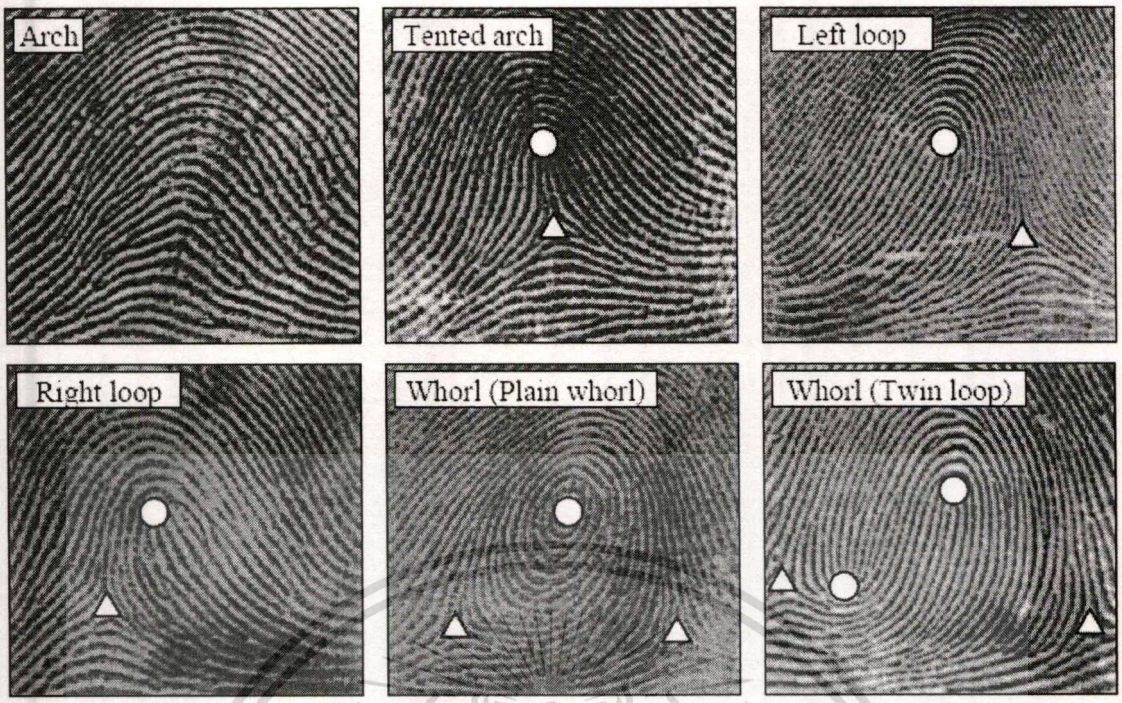


Figure 2.9 The five commonly used fingerprint classes.

The techniques fingerprint classification of problem have attracted a significant amount of interest in the scientific community due to its importance and intrinsic difficulty, and a large number of papers have been published on this topic during the last 30 years. Although a wide variety of classification algorithms has been developed for this problem, a relatively small number of features extracted from fingerprint images have been used by most of the authors.

The classification algorithm summarized here (see Figure 2.10) essentially devises a sequence of tests for determining the class of a fingerprint and conducts simpler tests earlier in the decision tree. For instance, two core points are typically detected for a whorl (see Figure 2.10) which is an easier condition to verify than detecting the number of Type-2 recurring ridges. Another highlight of the algorithm is that if does not detect the salient characteristics of any category from features detected in a fingerprint; it re-computes the features with a different pre-processing method. For instance, in the current implementation, the differential pre-processing consists of a different method/scale of smoothing. As can be observed from the flowchart that the algorithm detects *i)* whorls based upon detection of either two core points or a sufficient number of Type-2 recurring ridges; *ii)* arch based upon the inability to detect either delta or core points; *iii)* left (right) loops based on the characteristic tilt of the symmetric axis, detection of a core point, and detection of either a delta point or a sufficient number of Type-1 recurring curves; and

iv) tented arch based on relatively upright symmetric axis, detection of a core point, and detection of either a delta point or a sufficient number of Type-1 recurring curves. Figure 2.10 inset also illustrates ridge classification [37]. The "re-compute" option involves starting the classification algorithm with a different preprocessing (e.g., smoothing) of the image.

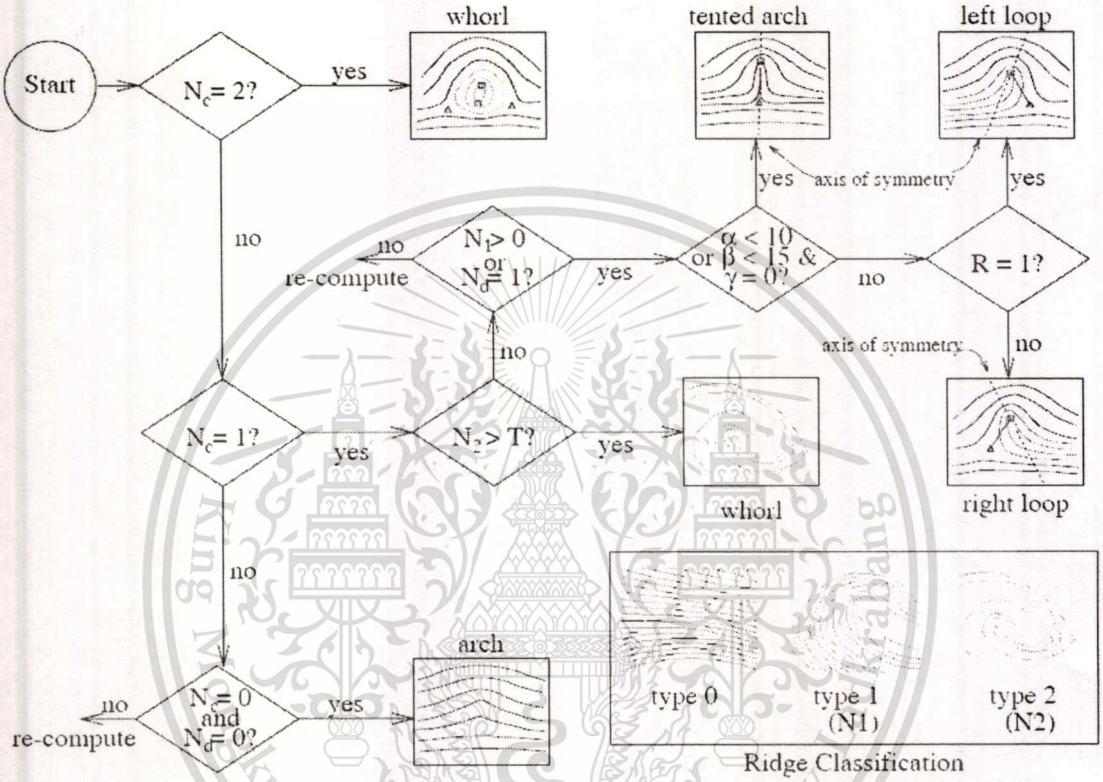


Figure 2.10 Flowchart of fingerprint classification algorithm.

In particular, almost all the methods are based on one or more of the following features (Figure 2.11): ridge line flow, orientation image, singular points, and Gabor filter responses. The ridge line flow is usually represented as a set of curves running parallel to the ridge lines; these curves do not necessarily coincide with the fingerprint ridges and valleys, but they exhibit the same local orientation.

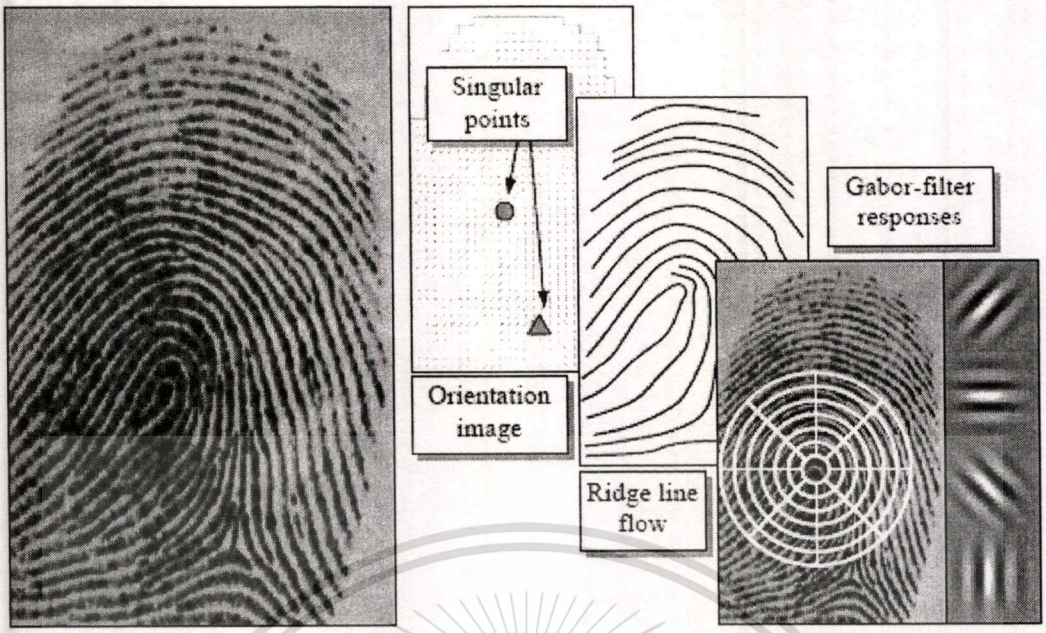


Figure 2.11 Most frequently used features for fingerprint classification.

Most of the existing fingerprint classification methods can be coarsely assigned to one of these categories: rule-based, syntactic, structural, statistical, neural network-based and multi-classifier approaches. Table 2.1 highlights the features used and the classification strategy adopted by fingerprint classification methods published over the last 30 years. Fingerprint classification is one of the most representative pattern recognition problems; by observing the table.

Table 2.1 Highlights the features used and the classification.

Fingerprint Classification Approach	Features				Classifier					
	O	S	R	G	Rb	Sy	Str	Sta	Nn	Mc
Moayer and Fu (1975)	✓					✓				
Moayer and Fu (1976)	✓					✓				
Rao and Balck (1980)	✓					✓				
Kawagoe and Tojo (1984)		✓	✓		✓					
Hughes and Green (1991)	✓								✓	
Bowen (1992)	✓	✓							✓	
Kamijo, Mieno, and Kojima (1992)	✓								✓	
Kamijo (1993)	✓								✓	
Moscinska and Tyma (1993)	✓				✓				✓	
Wilson, Candela, and Watson (1994)	✓								✓	
Candela et al. (1995)	✓		✓		✓				✓	✓
Omidvar, Blue, and Wilson (1995)	✓								✓	
Halici and Ongun (1996)	✓								✓	
Karu and Jain (1996)		✓			✓					
Maio and Maltoni (1996)	✓						✓			
Ballan, Sakarya, and Evans (1997)		✓			✓					
Chong et al. (1997)			✓		✓					
Senior (1997)			✓				✓			
Wei, Yuan, and Jie (1998)	✓				✓				✓	✓
Cappelli et al. (1999)	✓						✓			
Cappelli, Maio, and Maltoni (1999)	✓							✓		
Hong and Jain (1999)		✓	✓		✓					✓
Jain, Prabhakar, and Hong (1999)				✓				✓	✓	✓
Lumini, Maio, and Maltoni (1999)	✓						✓			
Cappelli, Maio, and Maltoni (2000a)	✓							✓		✓
Cho et al. (2000)		✓			✓					
Bartesaghi, and Gómez (2001)		✓			✓					

Fingerprint Classification Approach	Features				Classifier					
	O	S	R	G	Rb	Sy	Str	Sta	Nn	Mc
Bernard et al. (2001)	✓								✓	
Marcialis, Roli, and Frasconi (2001)	✓			✓			✓	✓	✓	✓
Pattichis et al. (2001)	✓				✓				✓	✓
Senior (2001)	✓		✓		✓		✓		✓	✓
Yao, Frasconi, and Pontil (2001)				✓				✓		✓
Cappelli, Maio, and Maltoni (2002a)	✓							✓		✓
Jain and Minut (2002)			✓		✓					
Cappelli et al. (2003)	✓							✓		✓
Yao et al. (2003)	✓			✓			✓	✓	✓	✓
Cappelli and Maio (2004)	✓							✓		✓
Klimanee and Nguyen (2004)	✓	✓			✓					
Senior and Bolle (2004)	✓		✓		✓		✓		✓	✓
Shah and Sastry (2004)								✓	✓	✓
Wang and Xie (2004)	✓	✓	✓		✓					
Zhang and Yan (2004)	✓	✓	✓		✓					
Park and Park (2005)	✓							✓		
Neuhaus and Bunke (2005)	✓						✓			
Tan, Bhanu, and Lin (2005)	✓							✓		
Min, Hong, and Cho (2006)				✓				✓		✓
Kristensen, Fyllingsnes (2007)				✓					✓	
Wang and Dai (2007)	✓	✓			✓					
Hong et al. (2008)	✓	✓		✓				✓		✓
Li, Yau, and Wang (2008)	✓	✓						✓		

As Table 2.1 chronological review of several fingerprints classification methods: each work is labeled according to the features used (O = orientation image, S = singularities, R = ridge flow, G = Gabor) and the classification technique (Rb = rule-based, Sy = syntactic, Str = structural, Sta = statistical, Nn = neural network, Mc = multiple classifiers)

2.3.1 Rule-based approaches

A fingerprint can be simply classified according to the number and the position of the singularities (see Table 2.2 and Figure 2.9); this is the approach commonly used by human experts for manual classification, therefore several authors proposed to adopt the same technique for automatic classification.

Table 2.2 Singular points in the five fingerprint classes.

Fingerprint class	Singular points
Arch	No singular points
Tented arch, Left loop, Right loop,	One loop and one delta
Whorl	Two loops (or a whorl) and two deltas

In [38] the Poincaré index is exploited to find type and position of the singular points and a coarse classification (according to Table 2.2) is derived. Then, a finer classification is obtained by tracing the ridge line flow. Discrimination among tented arch, left loop and right loop is performed according to the inclination of the central trace.

2.3.2 Syntactic approaches

A syntactic method describes patterns by means of terminal symbols and production rules; a grammar is defined for each class and a parsing process is responsible for classifying each new pattern [39]. The authors proposed a syntactic approach where terminal symbols are associated with small groups of orientation elements within the fingerprint orientation image; a class of context-free grammars is used to describe the fingerprint patterns, which are divided into seven classes [40]. The approach introduced by [41] is based on the analysis of ridge line flow, which is represented by a set of connected lines (Figure 2.12). These lines are labeled according to the direction changes, thus obtaining a set of strings that are processed through ad hoc grammars or string-matching techniques to derive the final classification.

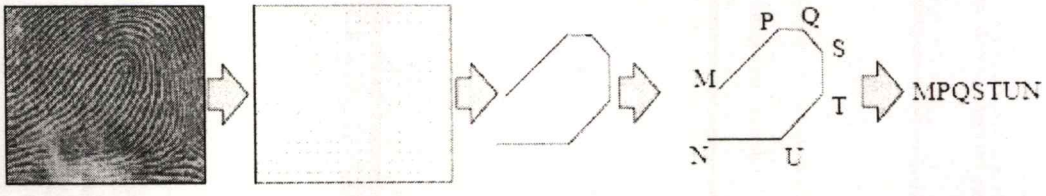


Figure 2.12 A schema of the string-construction approach.

2.3.3 Structural approaches

Structural approaches are based on the relational organization of low-level features into higher-level structures. This relational organization is represented by means of symbolic data structures, such as trees and graphs, which allow a hierarchical organization of the information [42]. The orientation image is well suited for structural representation: in fact, it can be partitioned into connected regions that are characterized by "homogeneous" orientations; these regions and the relations among them contain information useful for classification. This is the basic idea of the method proposed by [43]: the orientation image is partitioned into regions by minimizing a cost function that takes into account the variance of the element orientations within each region (Figure 2.13). A similar structural representation is also adopted by [44]. Another approach based on relational graphs, but created starting from different features, is proposed by [45]: regions that are relevant for the classification are extracted from the fingerprint orientation image through a directional variance filter that selects potential singular points and regions characterized by vertical orientations; the resulting structures are converted into attributed graphs and the classification is finally performed with a graph edit distance algorithm. Figure 2.13 Classification approach of [43]: from left to right: a fingerprint image, the partitioning of its orientation image, and the corresponding relational graph

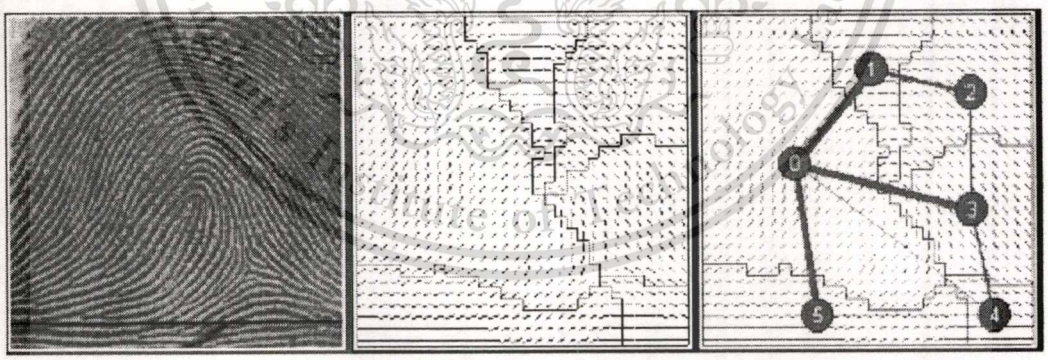


Figure 2.13 Classification approach.

2.3.4 Statistical approaches

In statistical approaches, a fixed-size numerical feature vector is derived from each fingerprint and a general-purpose statistical classifier is used for the classification. Some of the most widely adopted statistical classifiers [46] are: Bayes decision rule, *k*-nearest neighbor, and Support Vector Machines (SVM). Examples of their application can be found in several fingerprint-classification approaches, for instance:

- The Bayes decision rule is adopted by [47] to classify features learned by a genetic algorithm starting from the orientation image and by [48] to dynamically organize a set of SVM classifiers.
- The *k*-nearest neighbor is exploited in [49], where wedge-ring features obtained from the hexagonal Fourier transform are used as input, and in [50], where the first step of a two-stage classification technique is performed by means of the *k*-nearest neighbor rule.
- SVM classifiers are used by [51] to classify feature vectors obtained from the coefficients of a constrained nonlinear phase orientation model; SVM classifiers are often combined with other classifiers to improve the performance. Figure 2.14 in [52]: the templates corresponding to the five classes and an example of application of each template to the orientation image of a fingerprint belonging to the corresponding class.

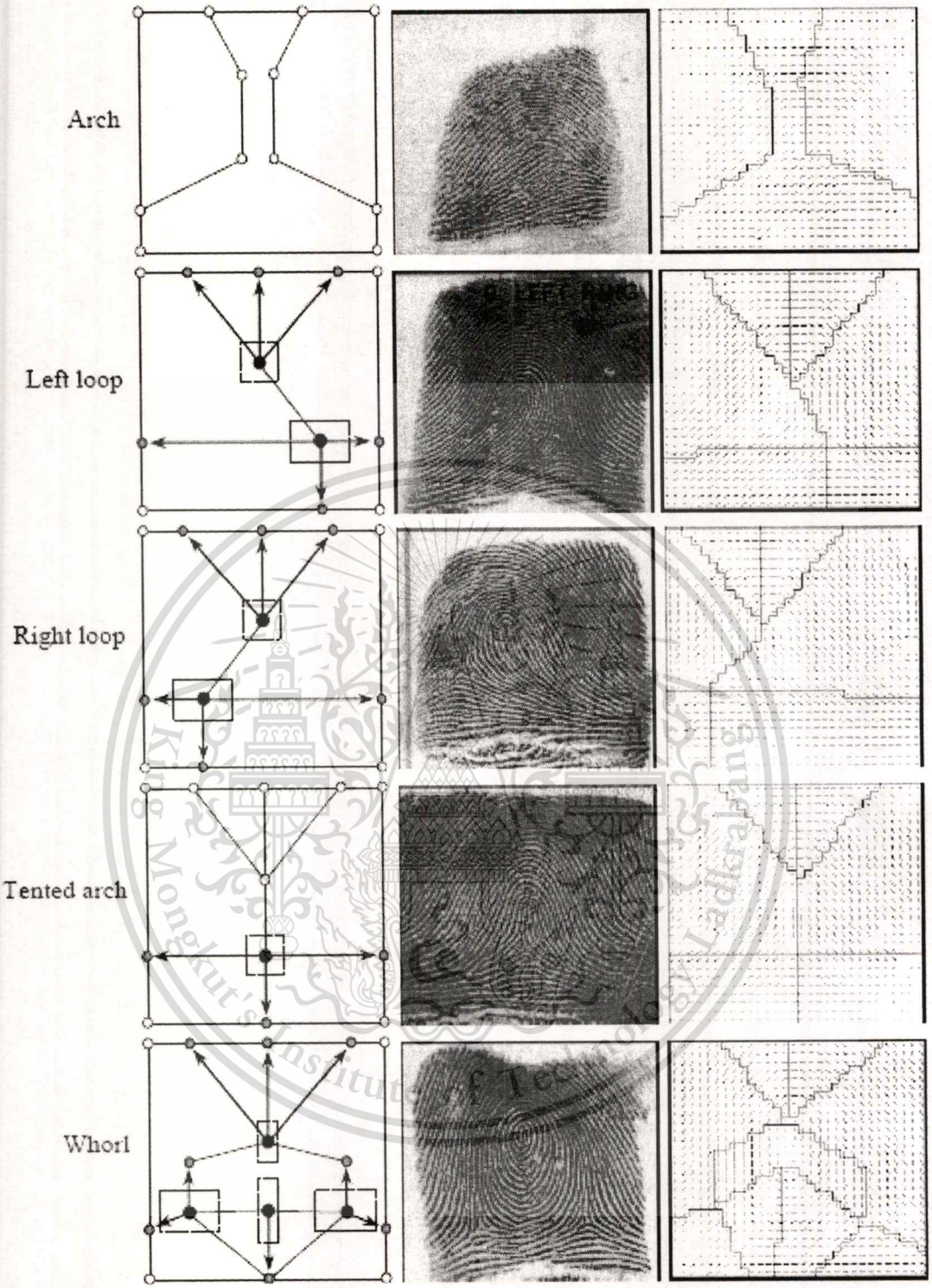


Figure 2.14 Classification scheme.

2.3.5 Neural network-based approaches

Most of the proposed neural network approaches are based on multilayer perceptions and use the elements of the orientation image as input features [53], the location of the singularities is used together with a 20 × 20 orientation image for the training of two disjoint neural networks, whose outputs are passed to a third one, which produces the final classification [54]. One of the best-known neural network approaches to fingerprint classification was proposed by NIST researchers [55].

2.3.6 Multiple classifier-based approaches

Different classifiers potentially offer complementary information about the patterns to be classified, which may be exploited to improve performance; in fact, in a number of pattern classification studies, it has been observed that different classifiers often misclassify different patterns. This motivates the recent interest in combining different approaches for the fingerprint classification task.

In [56] introduced PCASYS (Pattern-level Classification Automation SYStem): a complete fingerprint classification system based on the evolution of the methods proposed in. Figure 2.15 shows a functional schema of PCASYS: a probabilistic neural network which replaced the multilayer perceptions used in [55] is coupled with an auxiliary ridge tracing module, which determines the ridge flow in the bottom part of the fingerprint; this module is specifically designed to detect whorl fingerprints. Figure 2.15 in [56] of reprinted with permission from [52]

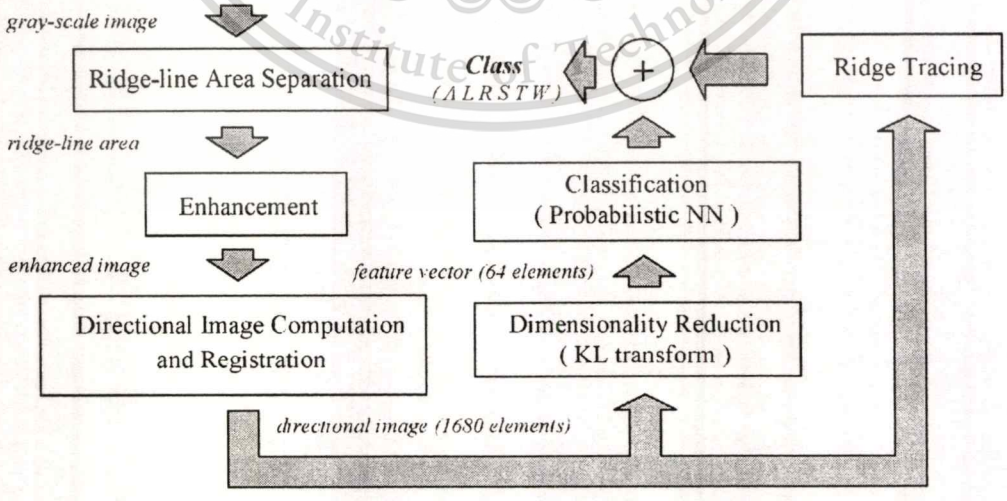


Figure 2.15 A functional scheme of PCASYS.

2.4 Techniques in Fingerprint Enhancement [1]

Among all the biometrics, fingerprint-based identification is one of the most popular and reliable biometric techniques. This is because it holds many desirable features such as universality, permanence, collectability, and distinctiveness. Personal identification based on fingerprint matching is now popular in wide range of applications. The most of fingerprint enhanced techniques [57-69]. The performance of minutiae extraction algorithms and other fingerprint recognition techniques relies heavily on the quality of the input fingerprint images. In an ideal fingerprint image, ridges and valleys alternate and flow in a locally constant direction. In such situations, the ridges can be easily detected and minutiae can be precisely located in the image. Figure 2.16 a) shows an example of a good quality fingerprint image. However, in practice, due to skin conditions (e.g., wet or dry, cuts, and bruises), sensor noise, incorrect finger pressure, and inherently low-quality fingers (e.g., elderly people, manual workers), a significant percentage of fingerprint images is of poor quality like those in Figures 2.16 b), c). In many cases, a single fingerprint image contains regions of good, medium, and poor quality where the ridge pattern is very noisy and corrupted (Figure 2.16). In general, there are several types of degradation associated with fingerprint images:

- The ridges are not strictly continuous; that is, the ridges have small breaks (gaps).
- Parallel ridges are not well separated. This is due to the presence of noise which links parallel ridges, resulting in their poor separation
- Cuts, creases, and bruises on the finger

These three types of degradation make ridge extraction extremely difficult in the highly corrupted regions. This leads to the following problems in minutiae extraction: (*i*) a significant number of spurious minutiae are extracted, (*ii*) a large number of genuine minutiae are missed, and (*iii*) large errors in the location (position and orientation) of minutiae are introduced. In order to ensure good performance of the ridge and minutiae extraction algorithms in poor quality fingerprint images, an enhancement algorithm to improve the clarity of the ridge structure is necessary. Figure 2.16 a) A good quality fingerprint; b) a medium quality fingerprint characterized by scratches and ridge breaks; c) a poor quality fingerprint containing a lot of noise.

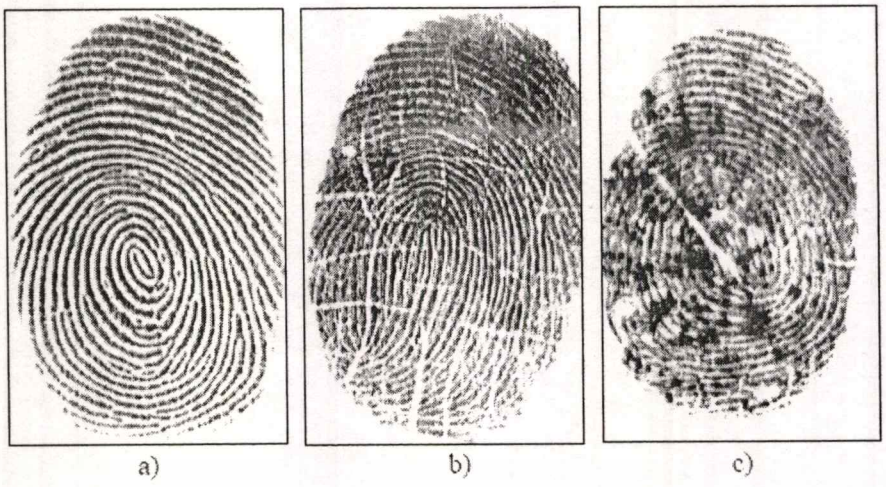


Figure 2.16 Example of different of quality fingerprint images.

A fingerprint expert is often able to correctly identify the minutiae by using various visual clues such as local ridge orientation, ridge continuity, ridge tendency, and so on. In theory, it is possible to develop an enhancement algorithm that exploits these visual clues to improve image quality. Generally, for a given fingerprint image, the fingerprint areas resulting from the segmentation step may be divided into three categories (Figure 2.17):

- Well-defined region: ridges can be clearly differentiated from each another.
- Recoverable region: ridges are corrupted by a small amount of gaps, creases, smudges, links, and the like, but they are still visible and the neighboring regions provide sufficient information about their true structure.
- Unrecoverable region: ridges are corrupted by such a severe amount of noise and distortion that no ridges are visible and the neighboring regions do not allow them to be reconstructed. Figure 2.17 a) a well-defined region; b) a recoverable region; c) an unrecoverable region

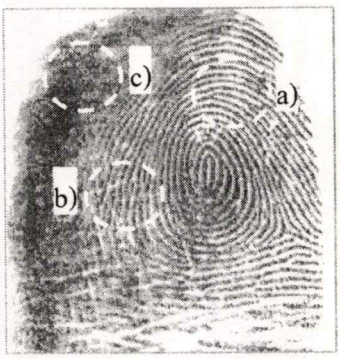


Figure 2.17 A fingerprint image containing regions of different quality.

Good quality regions, recoverable, and unrecoverable regions may be identified according to several criteria; in general, image contrast, orientation consistency, ridge frequency, and other local features may be combined to define a quality index. Since the estimation of fingerprint quality is central for a number of algorithms and practical applications, a section devoted to quality computation is provided at the end of this chapter. The goal of an enhancement algorithm is to improve the clarity of the ridge structures in the recoverable regions and mark the unrecoverable regions as too noisy for further processing. Usually, the input of the enhancement algorithm is a gray-scale image. The output may either be a gray-scale or a binary image, depending on the algorithm and goal.

2.4.1 Pixel-wise enhancement

In a pixel-wise image processing operation the new value of each pixel only depends on its previous value and some global parameters (but not on the value of the neighboring pixels). Pixel-wise techniques do not produce satisfying and definitive results for fingerprint image enhancement. However, contrast stretching, histogram manipulation; normalization used by [6, 57] determines the new intensity value of each pixel in an image as Figure 2.18 shows an example.

Figure 2.18 An example of normalization with the method described.

2.4.2 Contextual filtering

The most widely used technique for fingerprint image enhancement is based on contextual filters. In conventional image filtering, only a single filter is used for convolution throughout the image. In contextual filtering, the filter characteristics change according to the local context. Usually, a set of filters is pre-computed and one of them is selected for each image region. In fingerprint enhancement, the context is often defined by the local ridge orientation and

local ridge frequency. In fact, the sinusoidal-shaped wave of ridges and valleys is mainly defined by a local orientation and frequency that varies slowly across the fingerprint area. An appropriate filter that is tuned to the local ridge frequency and orientation can efficiently removes the undesired noise and preserve the true ridge and valley structure.

Several types of contextual filters have been proposed in the literature for fingerprint enhancement. Although they have different definitions, the intended behavior is almost the same: (1) provide a low-pass (averaging) effect along the ridge direction with the aim of linking small gaps and filling impurities due to pores or noise; (2) perform a bandpass (differentiating) effect in the direction orthogonal to the ridges to increase the discrimination between ridges and valleys and to separate parallel linked ridges.

Hong et al. [3, 57] proposed a fingerprint enhancement algorithm which decomposes the input fingerprint image into a set of filtered images. The orientation field and a quality mask which distinguishes recoverable and unrecoverable region are estimated from the set of filtered images. Gabor filters as bandpass filters are used to remove the noise and preserve true ridge/valley structures as Gabor filters have both frequency-selective and orientation selective properties and have optimal joint resolution in both spatial and frequency domains. For each input image eight Gabor filters with orientations 0°, 22.5° , 45° , 67.5° , 90° , 112.5° , 135° , 157.5° and radial bandwidth of 2.5 octaves is applied to obtain eight filtered images. Local orientation field is estimated for extracting ridges as; in the filtered images grey level values on ridges attain their local maxima along the direction that is orthogonal to local ridges. Then filter is applied to the image to adaptively accentuate the local maximum grey level values along the normal direction of the local ridge orientation. The ridges in the input fingerprint can be efficiently located by tuning the mask width to the width of the local ridges. For enhancement, area of each connected component appearing in the ridge map is computed and if area is less that a threshold then it is labeled as background otherwise, connected component is break into a set of short line segments. If each of short line segment is between a pair of narrow parallel ridges, it is labeled as true ridge otherwise background. After this most of the spurious ridges are removed.

In [58] proposed performed contextual filtering in the Fourier domain; in fact, it is well-known that a convolution in the spatial domain corresponds to a point by point complex multiplication in the Fourier domain [6]. The filter is defined in the frequency domain by the function:

$$H(\rho,\theta) = H_{radial}(\rho)H_{angle}(\theta), \tag{2.12}$$

where H_{radial} depends only on the local ridge spacing $\rho = 1/f$ and H_{radial} depends only on the local ridge orientation θ. Both H_{radial} and H_{radial} are defined as bandpass filters and are characterized by a mean value and a bandwidth. A set of n discrete filters is derived by their analytical definition. Actually, in the experiments, to reduce the number of filters, only a single value is used for the local ridge frequency and, therefore, the context is determined only by the orientation. The Fourier transform $P_i, i = 1 \ldots n$ of the filters is pre-computed and stored. Filtering an input fingerprint image $\boldsymbol{I}$ is performed as follows (see Figure 2.19) according to [58] the method.

- The FFT (Fast Fourier Transform) $\boldsymbol{F}$ of $\boldsymbol{I}$ is computed.
- Each filter P_i is point-by-point multiplied by $\boldsymbol{F}$, thus obtaining n filtered image transforms $PF_i, i = 1 \ldots n$ (in the frequency domain).
- Inverse FFT is computed for each PF_i resulting in n filtered images $PI_i, i = 1 \ldots n$ (in the spatial domain).

The enhanced image I_{enh} is obtained by setting, for each pixel (x, y), $I_{enh}(x,y) = PI_k(x,y)$, where k is the index of the filter whose orientation is the closest to θ_{xy}.

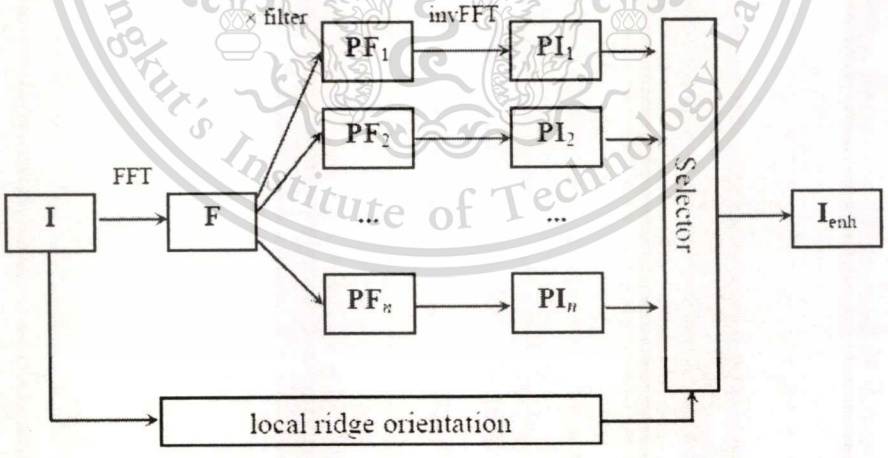


Figure 2.19 Enhancement of the fingerprint image.

Wang et al. [59] suggest replacing standard Gabor filter with Log-Gabor filter to overcome the drawbacks that the maximum bandwidth of a Gabor filter is limited to approximately one octave and Gabor filters are not optimal if one is seeking broad spectral

information with maximal spatial localization. For low-cost and computation-limited fingerprint systems (e.g., embedded systems), the 2D convolution of an image with a Gabor filter pre-computed over a discrete mask (e.g., 15 × 15) can be too time consuming. The computational complexity can be reduced by using separable Gabor filters.

In [60] proposed an efficient implementation of contextual filtering based on short-time Fourier transform (STFT) that requires partitioning the image into small overlapping blocks and performing Fourier analysis separately on each block. The orientation and frequency of each block are probabilistically determined. Each block is then filtered (by complex multiplication in the Fourier domain); in [58] the angular bandwidth is related to the distance from the closest singular point. Since singular point estimation is less robust than coherence estimation bandwidth adjustment seems to be more effective.

The output of a contextual fingerprint enhancement can be a gray-scale, near-binary, or binary image, depending on the filter parameters chosen. When selecting the appropriate set of filters and tuning their parameters, one should keep in mind that the goal is not to produce a good visual appearance of the image but to facilitate robustness of the successive feature extraction steps. If the filters are tuned to increase the contrast and suppress the noise, the estimation of the local context (orientation and frequency) may be erroneous in poor quality areas and the filtering is likely to produce spurious structures [61].

The need of an effective enhancement is particularly important in poor quality fingerprints where only the recoverable regions carry information necessary for matching. On the other hand, computing local information (context) with sufficient reliability in poor quality fingerprint images is very challenging. To overcome this problem [62, 63, 64] proposed to apply all the filters of a given set at each point in the image (as in Figure 2.19). A "selector" then chooses the best response from all the filter responses:

- In the method by [62], the selection is performed by minimizing an energy function that includes terms that require orientation and frequency to be locally smooth.
- In [63] base their selection on the analysis of local ridges extracted from the filtered images.
- In [64] make the selection according to the maximum response. However, unlike most of the Gabor-based methods, phase information coming from the real and the imaginary part of Gabor filters is also used for the final image enhancement.

As expected, approaches that require convolution of an image with a large number of filters are computationally expensive, and it is difficult to obtain efficient implementations.

2.4.3 Multi-resolution enhancement

Multi-resolution analysis has been proposed to remove noise from fingerprint images. Decomposing the image into different frequency bands (or sub-images) allows us to compensate for different noise components at different scales: in particular, at higher levels (low and intermediate frequency bands) the rough ridge-valley flow is cleaned and gaps are closed, whereas at the lower levels (higher frequencies) the finer details are preserved. The enhanced image bands are then recombined to obtain the final image. In [65] technique performs shape-adapted smoothing based on second moment descriptors and automatic scale selection (over a number of scales) based on normalized derivatives. The smoothing operation is adapted according to the local ridge structures, allowing interrupted ridges to be joined. The scale selection procedure estimates local ridge width and adapts the amount of smoothing to the local noise. In [66] the multi-resolution representation is based on wavelet decomposition. Each sub-image is processed through both textural and directional filtering to suppress spectral noise and to close the gaps produced by creases and scars. In [67] method is based on dyadic space scale and the decomposition depth is determined according to the average ridge width in the image; the noise reduction relies on smoothing the differential images between successive scales (i.e., the details that are lost passing from one scale to the successive one). In [68, 69] use a Laplacian like image-scale pyramid to decompose the original fingerprint into three smaller images corresponding to different frequency bands. Each image is then processed through contextual filtering.

2.5 Summary

Throughout this chapter several techniques for fingerprint matching, classification and enhancement have been surveyed and the pros and cons of different approaches have been highlighted. However, an explicit answer has not been provided to the question: what are the best algorithms for theses? The main reasons why it is difficult to assess the relative performance of the various these algorithms are as follows:

The performance of a fingerprint matching (recognition) method involves a tradeoff among different indicators: accuracy, efficiency, template size, and so on. Different applications have different performance requirements. For example, an application may prefer a fingerprint

matching algorithm that is lower in accuracy but has a small template size over an algorithm that is more accurate but requires a large template size; specific constraints are also imposed by system security related issues

Fingerprint classification has been the subject of several pattern recognition studies. Different solutions have been proposed and it is now possible to design classification systems that are able to meet the FBI requirement of highs performance .However, it is unlikely that exclusive classification would make it possible to significantly reduce the effort of searching for a single fingerprint in the absence of other information (e.g., sex, age, race, etc.).

Most of the Fingerprint enhancement was based on purpose fingerprints matching and classification techniques, therefore the resulting matching performance was often inadequate. Subsequently, several special purpose (domain specific) algorithms have been designed for processing fingerprint images, exploiting the peculiarity of fingerprint patterns: an example is contextual enhancement.

Chapter 3

General Techniques in Image Processing

3.1 Images and Digital Images [70]

Suppose we take an image, a photo, say. For the moment, let's make things easy and suppose the photo is black and white (that is, lots of shades of grey), so no color. We may consider this image as being a two dimensional function, where the function values give the brightness of the image at any given point, as shown in Figure 3.1. We may assume that in such an image brightness values can be any real numbers in the range 0 (black) to 1(white). The ranges of x and y will clearly depend on the image, but they can take all real values between their minima and maxima.

A digital image differs from a photo in that the x, y, and $f(x, y)$ values are all discrete. Usually they take on only integer values, so the image shown in Figure 3.1 will have x and y ranging from 1 to 256 each, and the brightness values also ranging from 0 (black) to 255 (white). A digital image can be considered as a large array of discrete dots, each of which has a brightness associated with it. These dots are called picture elements, or more simply pixels. The pixels surrounding a given pixel constitute its neighborhood. A neighborhood can be characterized by its shape in the same way as a matrix: we can speak of a 3x3 neighborhood, or of a 7x7 neighborhood. Except in very special circumstances, neighborhoods have odd numbers of rows and columns; this ensures that the current pixel is in the centre of the neighborhood. An example of a neighborhood is given in Figure 3.2, If a neighborhood has an even number of rows or columns (or both), it may be necessary to specify which pixel in the neighborhood is the "current pixel".

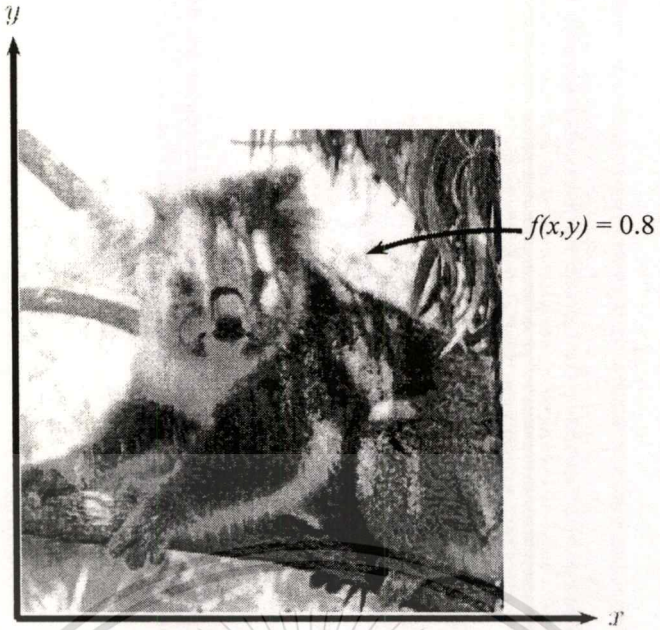


Figure 3.1 An image as a function

48	219	168	145	244	188	120	58
49	218	87	94	133	35	17	148
174	151	74	179	224	3	252	194
77	127	87	139	44	228	149	135
138	229	136	113	250	51	108	163
38	210	185	177	69	76	131	53
178	164	79	158	64	169	85	97
96	209	214	203	223	73	110	200

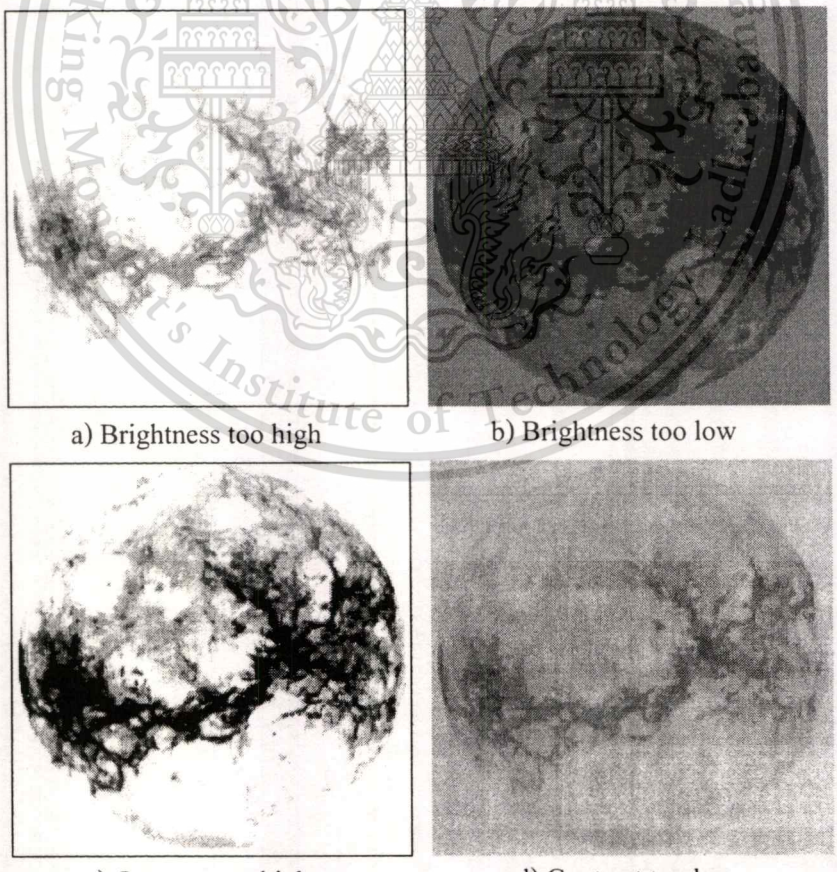
Current pixels

3x5 neighborhood

Figure 3.2 Pixels, with a neighborhood

3.2 Brightness and Contrast Adjustments

An image must have the proper brightness and contrast for easy viewing. Brightness refers to the overall lightness or darkness of the image. Contrast is the difference in brightness between objects or regions. For i.e., a white rabbit running across a snowy field has poor contrast, while a black against the same white background has good contrast. Figure 3.3 shows four possible ways that brightness and contrast can be misadjusted. When the brightness is too high, as in a), the whitest pixels are saturated, destroying the detail in these areas. The reverse is shown in b), where the brightness is set too low, saturating the blackest pixels. Figure 3.3 c) shows the contrast set to high, resulting in the blacks being too black, and the whites being too white. Lastly, d) has the contrast set too low; all of the pixels are a mid-shade of gray making the objects fade into each other. Brightness and contrast adjustments, increasing the brightness makes every pixel in the image becomes lighter. In comparison, increasing the contrast makes the light areas become lighter, and the dark areas become darker. These images show the effect of misadjusting the brightness and contrast



a) Brightness too high b) Brightness too low

c) Contrast too high d) Contrast too low

Figure 3.3 Brightness and contrast adjustments.

3.3 Histogram [6]

The histogram of the digital image with intensity levels in the range [0, L-1] is the discrete function $h(r) = n_k$, where, r_k is the k^{th} intensity value and n_k is the number of pixel in the image with intensity r_k. It is common practice to the normalized a histogram by dividing each of its components the total number of pixels in the image, denote by the product M N, where, as usual M and N are the row and column dimensions of the image. Thus, a normalized histogram given by $p(r_k)n_k / MN$, for k= 1, 2, 3.... L-1. Loosely speaking $p(r_k)$ is an estimate of the probability of occurrence of intensity level r_k in an image. The sum of all components of a normalized histogram is equal to 1. Histograms are the basis for numerous spatial domain processing techniques. Histogram manipulation can be for image enhancement, as shown in this section. In addition to providing useful image statistics, we shall see in subsequent chapters that the information inherent in histograms also in quite useful in order image processing and applications such as image compression and segmentation.

As an introduction to histogram processing for intensity transformation, consider Figure 3.4 shown in four basic intensity characteristics: low brightness (dark), high brightness (light), low contrast, and high contrast. The right side of four figures shows the histogram corresponding to these images. The horizontal axis of each histogram plot corresponds to intensity values, r_k. The vertical axis corresponds to values of $h(r) = n_k$ or $p(r_k)n_k / MN$ if the values are normalized. Thus, histograms may be viewed graphically simply as plots of $h(r) = n_k$ versus r_k or $p(r_k)n_k / MN$ versus r_k. We note in dark image that the components of the histogram are concentrated on the low (dark) side of the intensity scale. Similarly, the components of the histogram of the light image are biased toward the high side of the scale. An image with low contrast has a narrow histogram located typically toward the middle of the intensity scale. For monochrome image this implies a dull, washed-out gray look. Finally, we see that the components of the histogram in high-contrast image cover a wide range of the intensity scale and, future that the distribution of pixels is not too far from uniform, with very few vertical lines being much higher than the other.

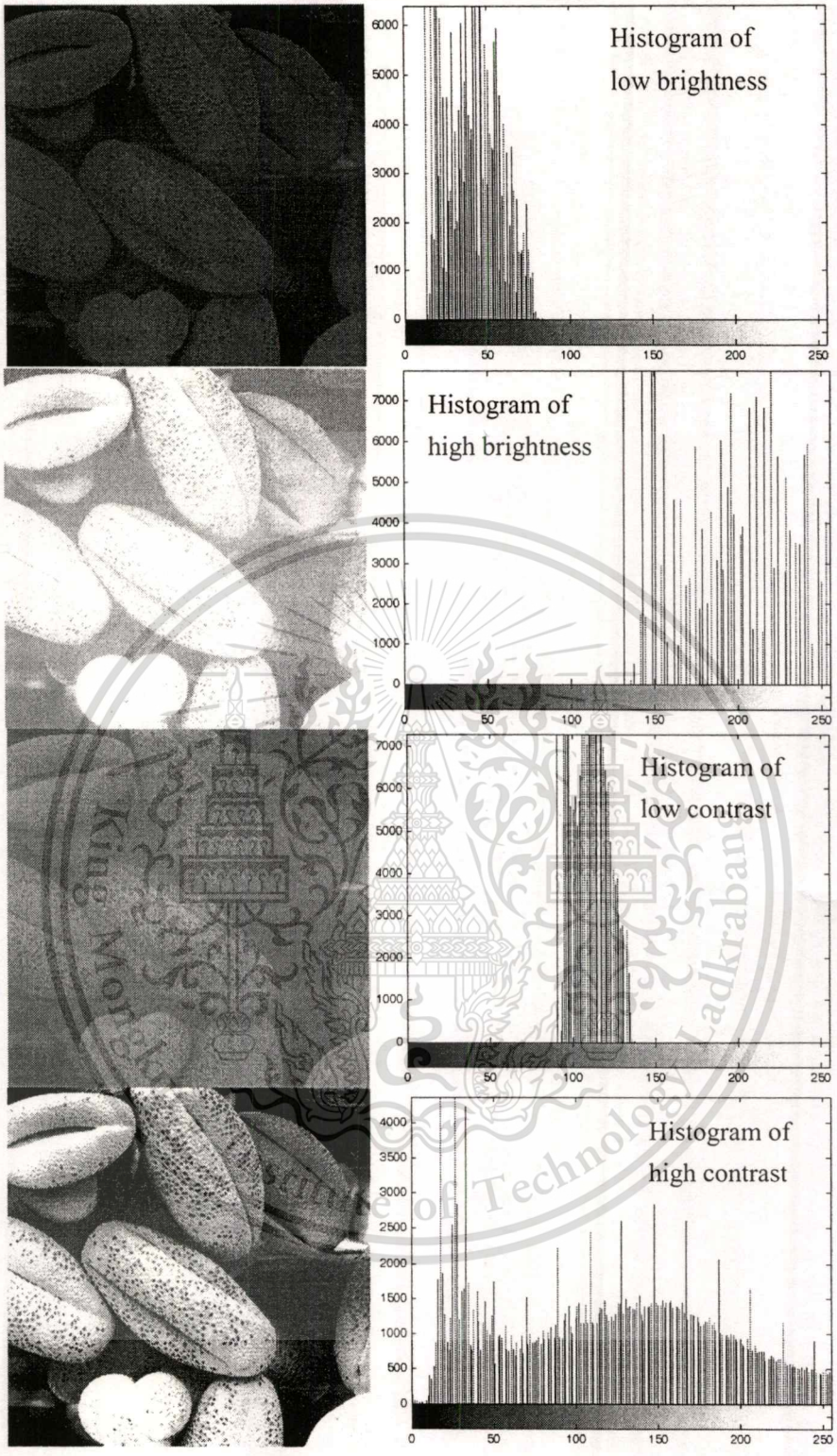


Figure 3.4 Four basic image types, low brightness, low contrast, high contrast, and their corresponding histograms.

3.4 Histogram Equalization [6]

Consider for moment continuous intensity values and let the variable r denote the intensity of image to be processed. As usual, we assume that r is in the range [0, L-1], with the $r=0$ representing block and r = L-1representing white. For r satisfying these conditions, we focus attention on transformations (intensity mapping) of the form that produce and output intensity level s for every pixel in the input image having intensity r. We assume that:

$$s = T(r) \qquad 0 \le r \le L-1 \tag{3.1}$$

(a) *T(r)* is a monotonically increasing function in the interval $0 \le r \le L-1$; and

(b) $0 \le T(r) \le L-1$ for $0 \le r \le L-1$. In some formulations to be discussed later, we used the inverse in which case we change condition (a) to

$$r = T^{-1}(s) \qquad 0 \le s \le L-1 \tag{3.2}$$

The requirement in conditional (a) that *T(r)* be monotonically increasing guarantees that output intensity values will never be less than corresponding input values, thus, preventing artifacts created by reversals of intensity.

For discrete values, we deal with probabilities (histogram values) and summations instead of probability density functions and integrals. As mentioned earlier, the probability of occurrence of level r_k in an image is approximated by

$$p_r(r_k) = \frac{n_k}{MN} \qquad k = 0,1,2,\cdots,L-1 \tag{3.7}$$

Where M N is the total number of pixel in the image; n_k is the number of pixels that have intensity r_k; and L is the number of possible intensity levels in the image (e.g., 256 for 8 bit image). The discrete from of the transformation in Eq. (3.4) is

$$\begin{aligned} s_k &= T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) \\ &= \frac{(L-1)}{MN}\sum_{j=0}^{k} n_j \qquad k = 0,1,2,\cdots,L-1 \end{aligned} \tag{3.8}$$

Thus, a processed (output) image is obtained by mapping each pixel in the input image with intensity r_k into a corresponding pixel with level s_k in the output image using Eq. (3.8). The

transformation (mapping) $T(r)$ in this equation is called histogram equalization or histogram linearization transformation.

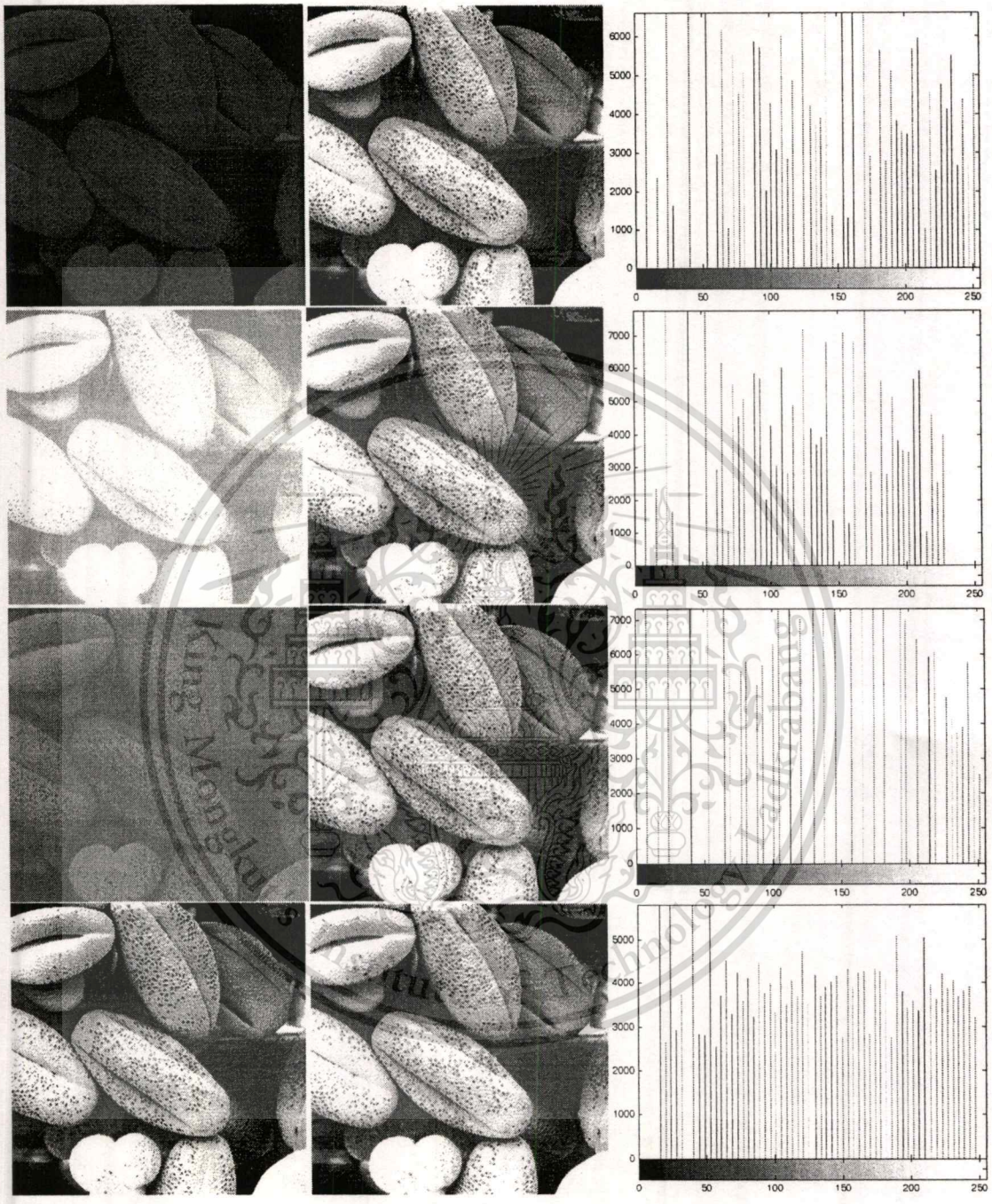


Figure 3.5 Left column: images from Figure 3.4. Center column: corresponding histogram equalized images. Right column: histograms of the images in the center column.

3.5 Normalization

In image processing, normalization is a process that changes the range of pixel intensity values. Applications include photographs with poor contrast due to glare, for example: normalization is sometimes called contrast stretching. In more general fields of data processing, such as digital signal processing, it is referred to as dynamic range expansion. Normalization is a linear process. If the intensity range of the image is 50 to 180 and the desired range is 0 to 255 the process entails subtracting 50 from each of pixel intensity, making the range 0 to 130 then each pixel intensity is multiplied by 255/130, making the range 0 to 255. The normalization process will produce iris regions, which have the same constant dimensions, so that two photographs of the same iris under different conditions will have characteristic features at the same spatial location. The processing of fingerprint normalization can reduce the variance in gray-level values along ridges and valleys by means of adjust the gray-level values to the predefined constant mean and variance. And normalization can remove the influences of sensor noise and gray-level deformation [3, 57, 77].

a) Original image b) Normalization

Figure 3.6 The fingerprint image of normalization.

3.6 Gaussian Filters

We have seen some examples of linear filters so far: the averaging filter, and a high pass filter. The *fspecial* function can produce many different filters for use with the *filter2* function; we shall look at a particularly important filter here. Gaussian filters are a class of low-pass filters, all based on the Gaussian probability distribution function

$$f(x) = e^{-\frac{x^2}{2\sigma^2}} \tag{3.9}$$

Where σ is the standard deviation, a large value of σ produces to a flatter curve, and a small value leads to a "pointier" curve. Figure 3.7 shows examples of such one dimensional Gaussians.

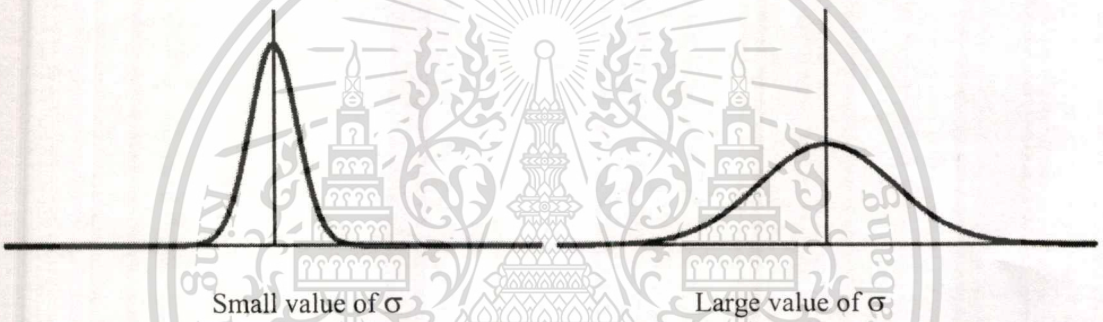


Figure 3.7 One dimensional Gaussians.

A two dimensional Gaussian function is given by

$$f(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{3.10}$$

I can shape of Gaussian this with the surf function, and to ensure a nice smooth result, we will create a large filter (size 50x50) with different standard deviations. Gaussian filters have a blurring effect which looks very similar to that produced by neighborhood averaging. Let's experiment with the fingerprint image, and some different Gaussian filters. The final parameter is the standard deviation; which if not given defaults to 3 and 9. The second parameter (which is also optional), gives the size of the filter; the default are 15x15 and 31x31

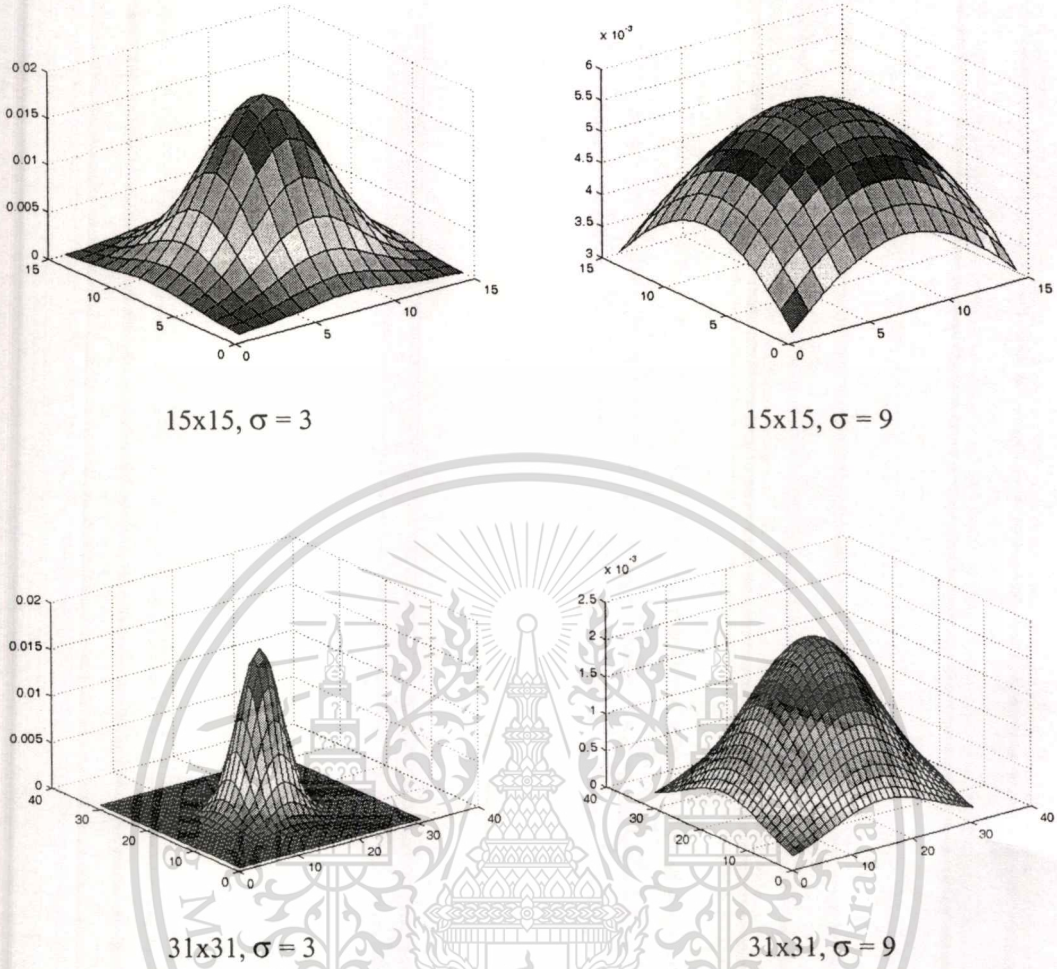


Figure 3.8 Two dimensional Gaussians.

We see that to obtain a spread out blurring effect, we need a large standard deviation. In fact, if we let the standard deviation grow large without bound, we obtain the averaging filters as limiting values. For example we have the 31x31 averaging filter. Although the results of Gaussian blurring and averaging look similar, the Gaussian filter has some elegant mathematical properties which make it particularly suitable for blurring.

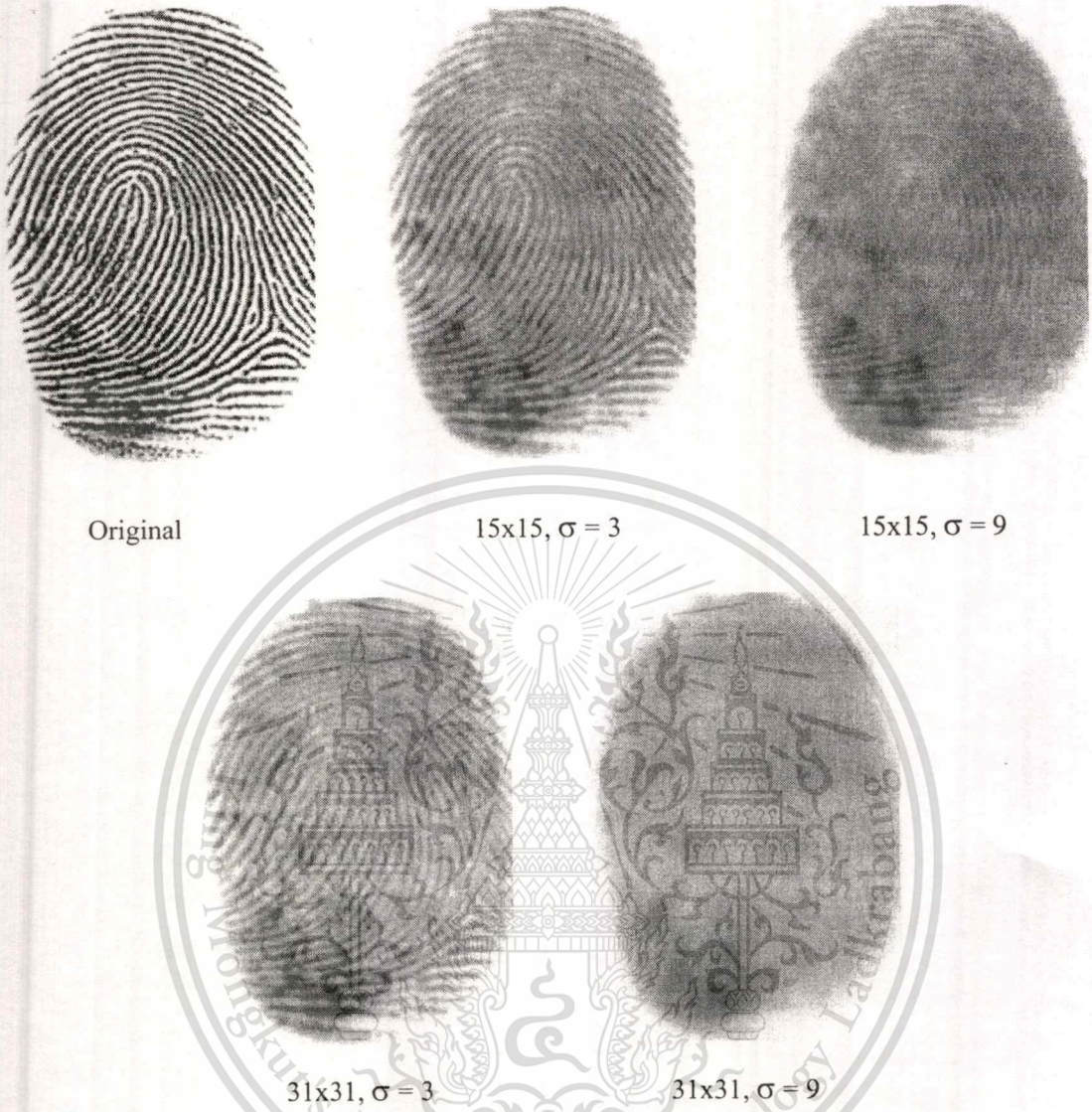


Figure 3.9 Effects of different Gaussian filters on an image.

3.7 Median Filtering

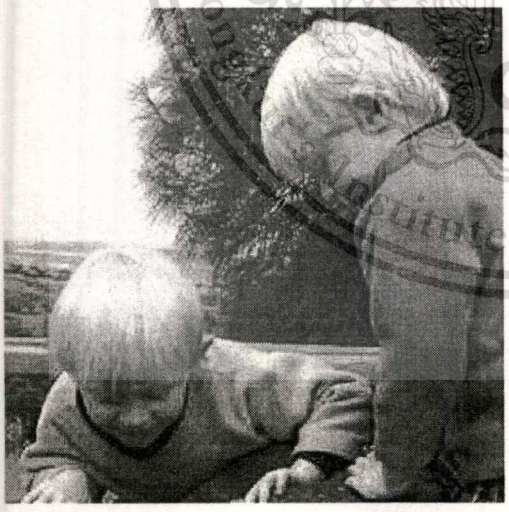
Median filtering seems almost tailor-made for removal of salt and pepper noise. Recall that the median of a set is the middle value when they are sorted. If there are even numbers of values, the median is the mean of the middle two. A median filter is an example of a non-linear spatial filter; using a 3x3 mask, the output value is the median of the values in the mask as Table3.1.

Table 3.1 3x3 mask of pixels.

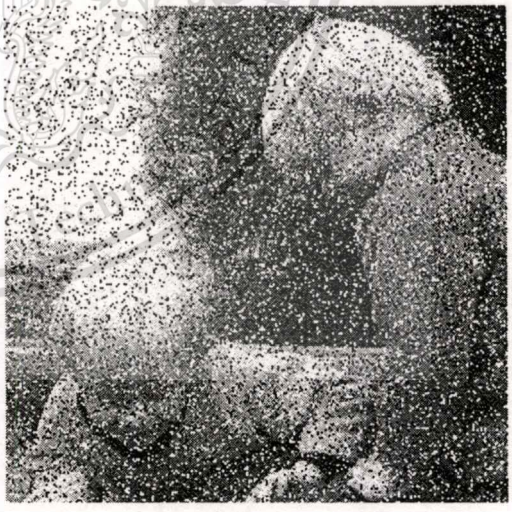
50	55	52
63	255	58
61	60	57

→ 50 52 57 58 **60** 61 63 65 255 → 60

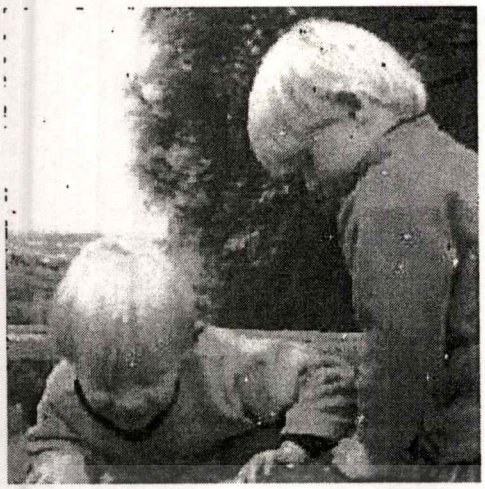
The operation of obtaining the median means that very large or very small values-noisy value- will end up at the top or bottom of the sorted list. Thus the median will in general replace a noisy value with one closer to its surroundings.



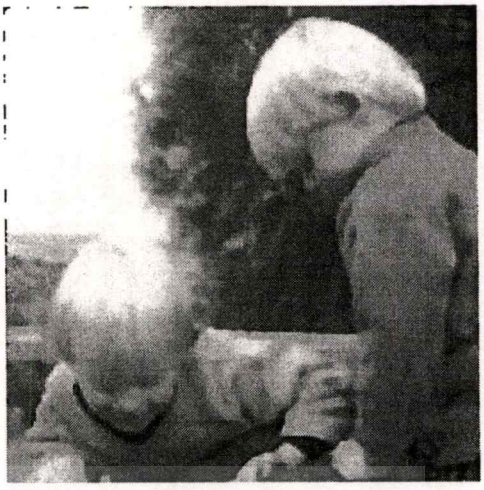
a) Original image



b) 20% salt & pepper noise



c) Using a 3x3 median filtering



b) Using a 5x5 median filter

Figure 3.10 Cleaning 20% salt & pepper noise with median filtering.

3.8 Edge Detection

Edges contain some of the most useful information in an image. We may use edges to measure the size of objects in an image; to isolate particular objects from their background; to recognize or classify objects. There are a large number of edge-finding algorithms in existence, and we shall look at some of the more straightforward of them. An edge may be loosely defined as a local discontinuity in the pixel values which exceeds a given threshold. More informally, an edge is an observable difference in pixel values. For example, consider the two 4 blocks of pixels shown in Table 3.2.

Table 3.2 4x4 blocks of pixels

51	52	53	59
54	52	53	62
50	52	53	68
55	52	53	55

50	53	155	160
51	53	160	170
52	53	167	190
51	53	162	155

In the right hand block, there is a clear difference between the grey values in the second and third columns, and for these values the differences exceed 100. This would be easily discernible in an image the human eye can pick out grey differences of this magnitude with relative ease. Our aim is to develop methods which will enable us to pick out the edges of an image.

3.8. 1 Derivatives and edges

Consider the image in Figure 3.11, and suppose we plot the gray values as we traverse the image from left to right. Two types of edges are illustrated here: a ramp edge, where the grey values change slowly, and a step edge, or an ideal edge, where the grey values change suddenly.

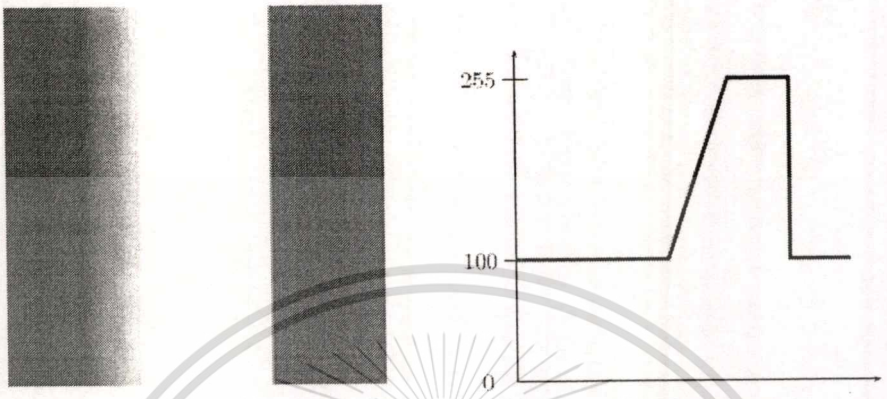


Figure 3.11 Edges and their profiles.

Suppose the function which provides the profile in Figure 3.11 is *f(x)*; then its derivative *f′(x)* can be plotted; this is shown in Figure 3.12 The derivative, as expected, returns zero for all constant sections of the profile, and is non zero (in this example) only in those parts of the image in which difference occur.

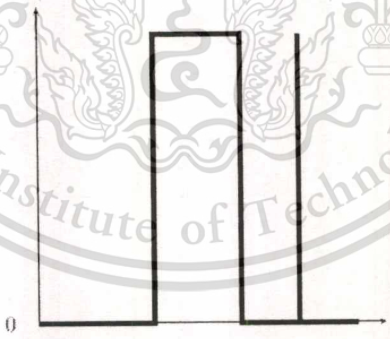


Figure 3.12 The derivative of the edge profile.

Many edge finding operators are based on differentiation; to apply the continuous derivative to a discrete image, first recall the definition of the derivative:

$$\frac{df}{dx} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h} \tag{3.11}$$

Since in an image, the smallest possible value of h is 1, being the difference between the index values of two adjacent pixels, a discrete version of the derivative expression is

$$f(x+1)-f(x) \tag{3.12}$$

other expressions for the derivative are

$$\lim_{h\to 0}\frac{f(x)-f(x-h)}{h},\quad \lim_{h\to 0}\frac{f(x+h)-f(x-h)}{2h} \tag{3.13}$$

with discrete counterparts

$$f(x)-f(x-1),\quad \big(f(x+1)-f(x-1)\big)/2 \tag{3.14}$$

for an image, with two dimensions, we use partial derivatives; an important expression is the gradient, which is the vector defined by

$$\begin{bmatrix}\dfrac{\partial f}{\partial x} & \dfrac{\partial f}{\partial y}\end{bmatrix} \tag{3.15}$$

which for a function *f(x, y)* points in the direction of its greatest increase. The direction of that increase is given by

$$\tan^{-1}\left(\frac{\partial f/\partial y}{\partial f/\partial x}\right) \tag{3.16}$$

and its magnitude by

$$\sqrt{\left(\frac{\partial f}{\partial x}\right)^2+\left(\frac{\partial f}{\partial y}\right)^2} \tag{3.17}$$

Most edge detection methods are concerned with finding the magnitude of the gradient, and then applying a threshold to the result. Some edge detection filters using the expression $f(x+1)-f(x-1)$ for the derivative, leaving the scaling factor out, produces horizontal and vertical filters:

$$\begin{bmatrix}-1 & 0 & 1\end{bmatrix} \quad \text{and} \quad \begin{bmatrix}-1\\0\\1\end{bmatrix} \tag{3.18}$$

These filters will find vertical and horizontal edges in an image and produce a reasonably bright result. However, the edges in the result can be a bit "jerky"; this can be overcome by smoothing the result in the opposite direction; by using the filters

$$\begin{bmatrix}1\\1\\1\end{bmatrix} \quad \text{and} \quad \begin{bmatrix}1 & 1 & 1\end{bmatrix} \tag{3.19}$$

1) The Prewitt filters.

This filter and its companion for finding vertical and horizontal edges:

$$P_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad \text{and} \quad P_x = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \tag{3.20}$$

2) The Roberts cross-gradient filters

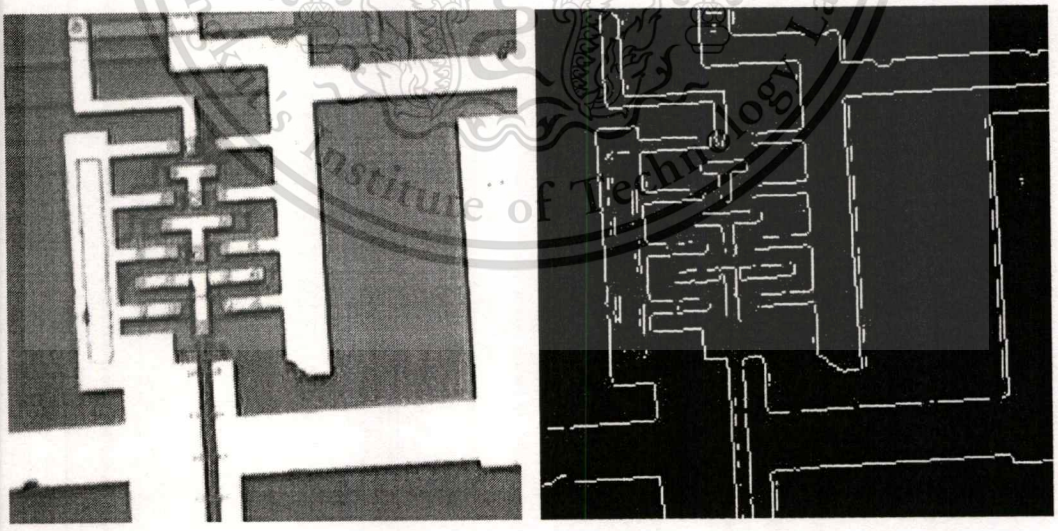
$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.21}$$

3) The Sobel filters

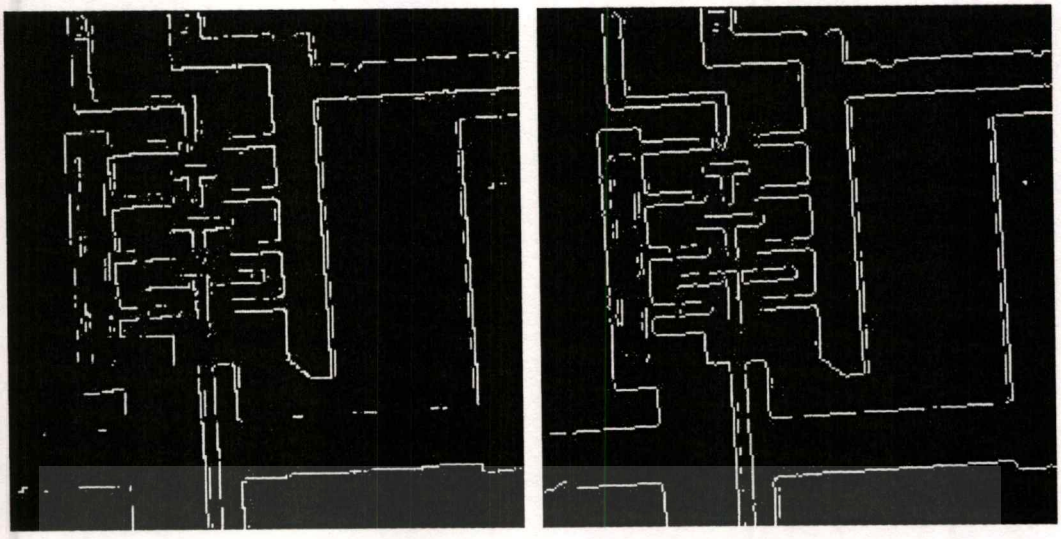
$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} -1 & -2 & 1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{3.22}$$

The Sobel filters are similar to the Prewitt filters, in that they apply a smoothing filter in the opposite direction to the central difference filter. In the Sobel filters, the smoothing takes the form

$$\begin{bmatrix} 1 & 2 & 1 \end{bmatrix} \tag{3.23}$$



a) Integrated circuit b) Prewitt



c) Roberts d) Sobel

Figure 3.13 Results of the edge detections.

3.8.2 Second derivatives

1) The Laplacian

Another class of edge-detection method is obtained by considering the second derivatives. The sum of second derivatives in both directions is called the laplacian; it is written as

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{^2 f}{\partial y^2} \tag{3.24}$$

and it can be implemented by the filter

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{3.25}$$

This is known as a discrete Laplacian. The laplacian has the advantage over first derivative methods in that it is an isotropic filter; this means it is invariant under rotation. That is, if the laplacian is applied to an image, and the image then rotated, the same result would be obtained if the image was rotated first, and the laplacian applied second. This would appear to make this class of filters ideal for edge detection. However, a major problem with all second derivative filters is that they are very sensitive to noise. To see how the second derivative affects an edge, take the derivative of the pixel values as plotted in Figure 3.11; the results are shown schematically in Figure 3.14.

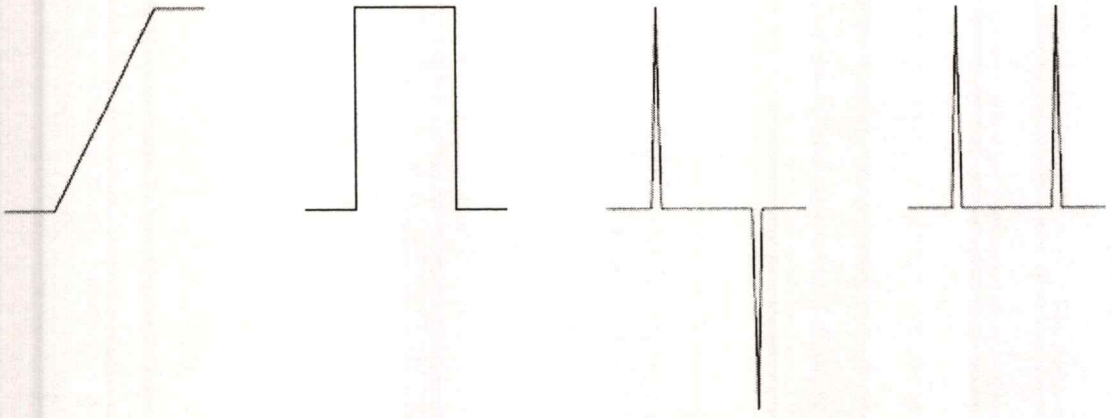


Figure 3.14 Second derivatives of an edge function.

Although the result is adequate and shown Figure 3.15, it is very messy when compared to the results of the Prewitt and Sobel methods discussed earlier. Other Laplacian masks can be used; some are:

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} -2 & 1 & -2 \\ 1 & 4 & 1 \\ -2 & 1 & -2 \end{bmatrix} \tag{3.26}$$

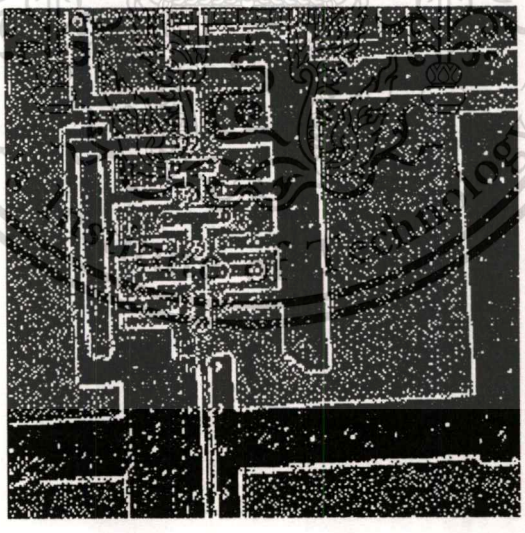


Figure 3.15 Result after filtering with a discrete laplacian.

2) Zero crossings

A more appropriate use for the Laplacian is to find the position of edges by locating zero crossings. From Figure 3.14, the position of the edge is given by the place where the value of the filter takes on a zero value. In general, these are places where the results of the filter changes sign. For example, consider the simple image given in Table 3.3 a), and the result after filtering with a Laplacian mask in Table 3.3 b).

Table 3.3 Locating zero crossings in an image.

50	50	50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50	50	50
50	50	200	200	200	200	200	200	50	50
50	50	200	200	200	200	200	200	50	50
50	50	200	200	200	200	200	200	50	50
50	50	200	200	200	200	200	200	50	50
50	50	50	50	200	200	200	200	50	50
50	50	50	50	200	200	200	200	50	50
50	50	50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50	50	50

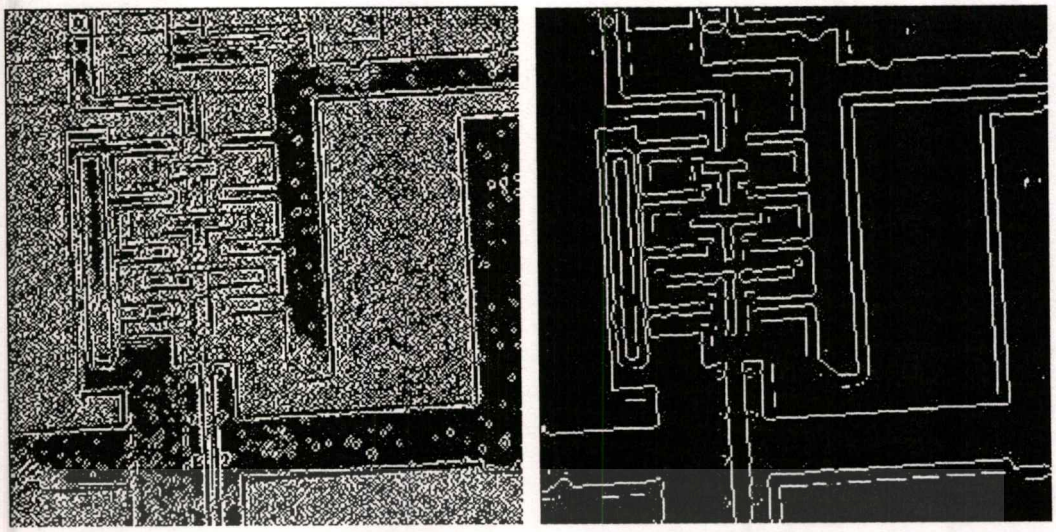
a) A simple image

−100	−50	−50	−50	−50	−50	−50	−50	−50	−100
−50	0	150	150	150	150	150	150	0	−50
−50	150	−300	−150	−150	−150	−150	−300	150	−50
−50	150	−150	0	0	0	0	−150	150	−50
−50	150	−150	0	0	0	0	−150	150	−50
−50	150	−300	−150	0	0	0	−150	150	−50
−50	0	150	300	−150	0	0	−150	150	−50
−50	0	0	150	−300	−150	−150	−300	150	−50
−50	0	0	0	150	150	150	150	0	−50
−100	−50	−50	−50	−50	−50	−50	−50	−50	−100

b) After laplace filtering

The result is shown in Figure 3.16 a) this is not in fact a very good result far too many grey level changes have been interpreted as edges by this method. To eliminate them, we may first smooth the image with a Gaussian filter. This leads to the following sequence of steps for edge detection: Smooth the image with a Gaussian filter; sonvolve the result with a laplacian and find the zero crossings

This method was designed to provide a edge detection method to be as close as possible to biological vision. The first two steps can be combined into one, to produce a "Laplacian of Gaussian" or "LoG" filter. These filters can be created with the fspecial function. In fact the LoG and zerocross options implement the same edge finding method; the difference being that the zerocross option allows you to specify your own filter. The result after applying a LoG filter and finding its zero crossings is given in Figure 3.16 b)



a) Zeros crossings b) Using a LoG filter first

Figure 3.16 Edge detection by using zero crossing.

3.9 Some Applications of Images

Image processing has an enormous range of applications; almost every area of science and technology can make use of image processing methods. Here is a short list just to give some indication of the range of image processing applications.

Medicine: Inspection and interpretation of images obtained from X-rays, MRI or CAT scans, analysis of cell images, of chromosome karyo types.

Agriculture: Satellite/aerial views of land, for example to determine how much land is being used for different purposes, or to investigate the suitability of different regions for different crops, inspection of fruit and vegetables distinguishing good and fresh produce from old.

Industry: Automatic inspection of items on a production line, inspection of paper samples.

Law enforcement: Fingerprint analysis, sharpening or de-blurring of speed-camera images.

3.10 The Fast Wavelet Transform Algorithm [71]

The fast wavelet transform (FWT) is a computationally efficient implementation of the discrete wavelet transform (DWT) that exploits a surprising but fortunate relationship between coefficients of DWT adjacent scale but also called Mallat algorithm [6,72].

3.10.1 Four filters used to calculate the DWT and IDWT

For an orthogonal wavelet ψ, the associated scaling function φ satisfies a fundamental relation, which is the following twin scale equation:

$$\frac{1}{2}\varphi\left(\frac{t}{2}\right)=\sum_{n\in\mathbb{Z}}a_n\varphi_{0,n}=\sum_{n\in\mathbb{Z}}a_n\varphi(t-n) \tag{3.27}$$

The filters involved in the discrete wavelet transform (DWT)and in the inverse transform (IDWT) are closely linked to the $(a_n)_{n\in\mathbb{Z}}$ sequence. If φ (and, consequently, ψ) have compact support, the sequence $(a_n)_{n\in\mathbb{Z}}$ has only a finite number of non-zero elements. We may then see this sequence as a low-pass filter. This filter, noted w , thus lets the low frequencies through and retains the high ones. It has a finite impulse response (FIR), with length noted K , its sum is equal to 1 and its norm is $\frac{1}{\sqrt{2}}$.

Using the filter w we define four filters with finite impulse response, size K and norm 1. The decomposition filters are noted (indicated by the final *D*): *LoD* and *HiD*. The first is low-pass (indicated by the initial *Lo*) and the second is high-pass (indicated by the initial *Hi*). The two reconstruction filters (indicated by the final *R*) are noted *LoR* and *HiR*.

Table 3.4 Four filters

Filters	Low-pass	High-pass
Decomposition	*LoD*	*HiD*
Reconstruction	*LoR*	*HiR*

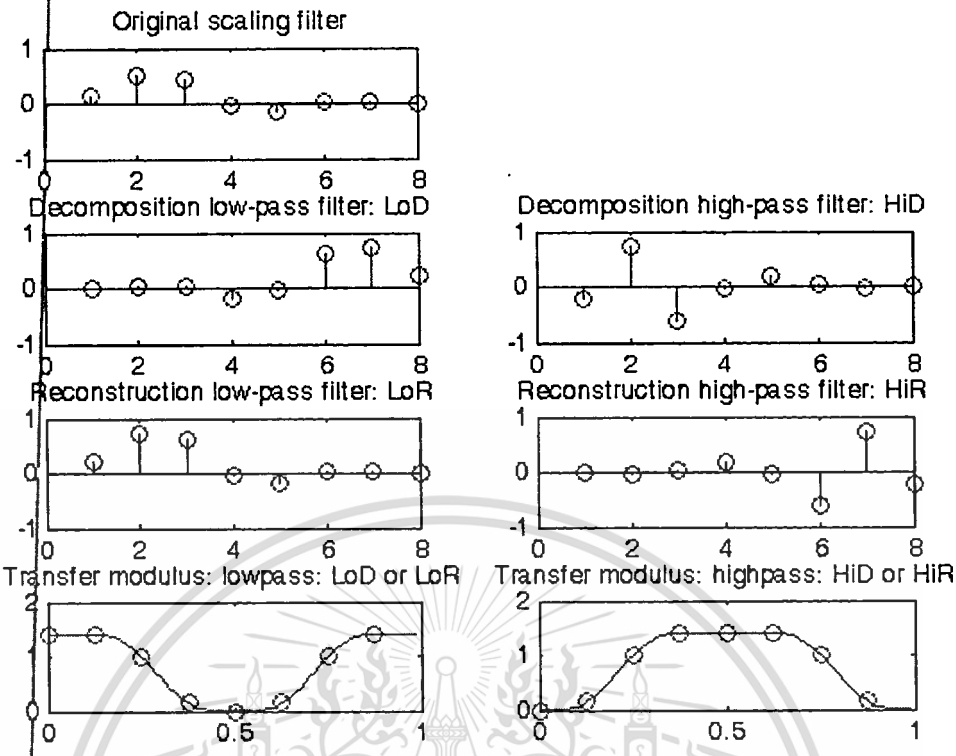


Figure 3.17 Filters for the wavelet *db4*

The two reconstruction filters are linked by:

$$LoR = \frac{w}{\|w\|} \text{ and } HiR_k = (-1)^k LoR_{K+1-k} \quad \text{for k = 1, 2,....., K} \qquad (3.28)$$

They are mirror filters in quadrature. The two decomposition filters are obtained via mirror image of the reconstruction filters:

$$LoD = LoR_{K+1-k} \text{ and } HiD_k = HiR_{K+1-k} \quad \text{for k = 1, 2,....., K} \qquad (3.29)$$

For the *db4* wavelet as below we represent in Figure 3.20: the filter corresponding to the scaling function, at the top; the four filters deduced from it, in the middle; and modules of the filters transfer functions, at the bottom.

3.10.2. Orthogonal wavelets

The strategy of construction of orthogonal wavelets with compact support motivated by the use of the fast Mallat algorithm is based on the direct action on the filter m_0, which generates the scaling function. The work of Daubechies at the end of the 1980s and at the start of the 1990s has marked a decisive stage in the diffusion of wavelet methods.

This family of wavelets with one parameter, due to Daubechies *dbN* is the first one to make it possible to handle orthogonal wavelets with compact support and arbitrary regularity. We will call N the order of the *dbN* wavelet. This family contains the *Haar* wavelet, *db1*, which is the simplest and certainly the oldest of wavelets. It is discontinuous, resembling a square form.

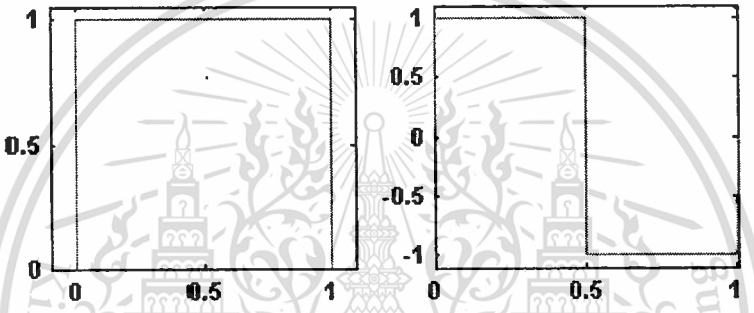


Figure 3.18 The scaling function (on the left) and the wavelet (on the right): Haar

The Haar wavelet is defined by: $\psi(x) = 1$ if $x \in [\,0, 0.5[$, $\psi(x) = -1$ if $x \in [\,0.5,1[$ and 0 if not. The associated scaling function is the function: $\varphi(x) = 1$ if $x \in [\,0,1]$ and 0 if not.

Except for *db1*, the wavelets of this family do not have an explicit expression. However, the square modulus of the transfer function of the associated filter h is explicit and relatively simple let:

$$P(y) = \sum_{k=0}^{N-1} C_{N-1+k}^{k} y^{k} \text{ , where } C_{N-1+k}^{k} \text{ are the binomial coefficients,} \tag{3.30}$$

$$|m_0(\omega)|^2 = \left(\cos^2\left(\frac{\omega}{2}\right)\right)^N P\left(\sin^2\left(\frac{\omega}{2}\right)\right), \text{ where } m_0(\omega) = \frac{1}{\sqrt{2}} \sum_{k=0}^{2N-1} h_k e^{ik\omega} \tag{3.31}$$

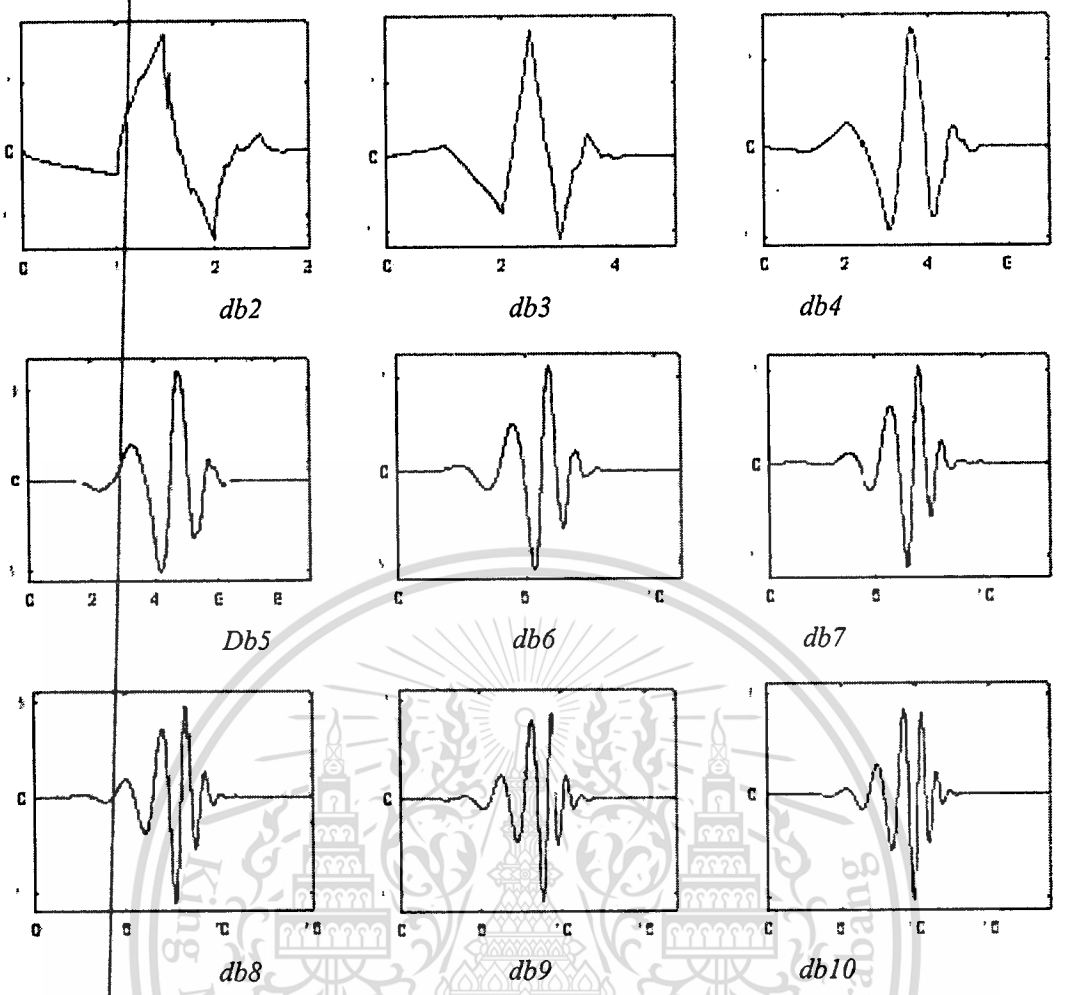
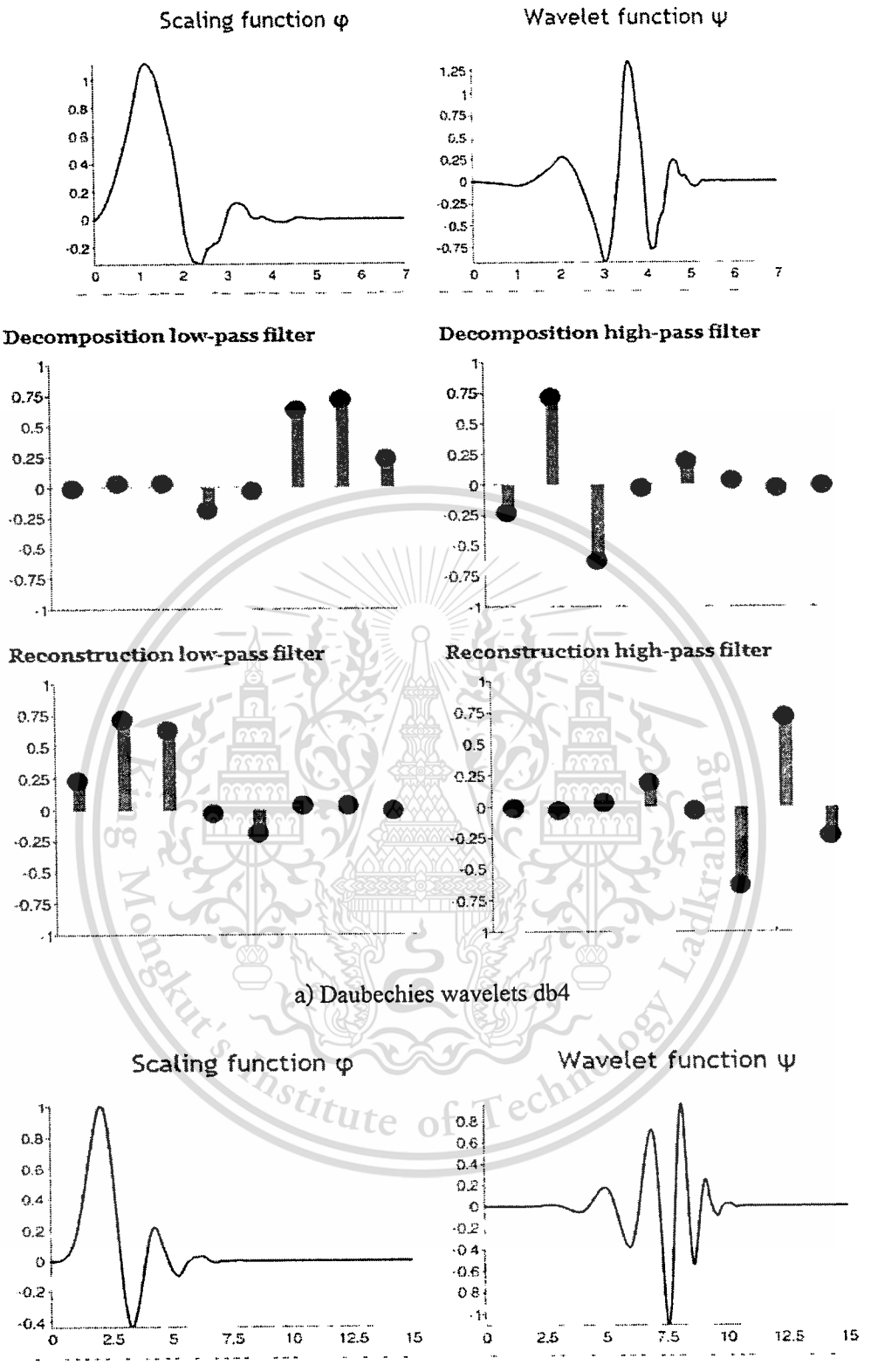


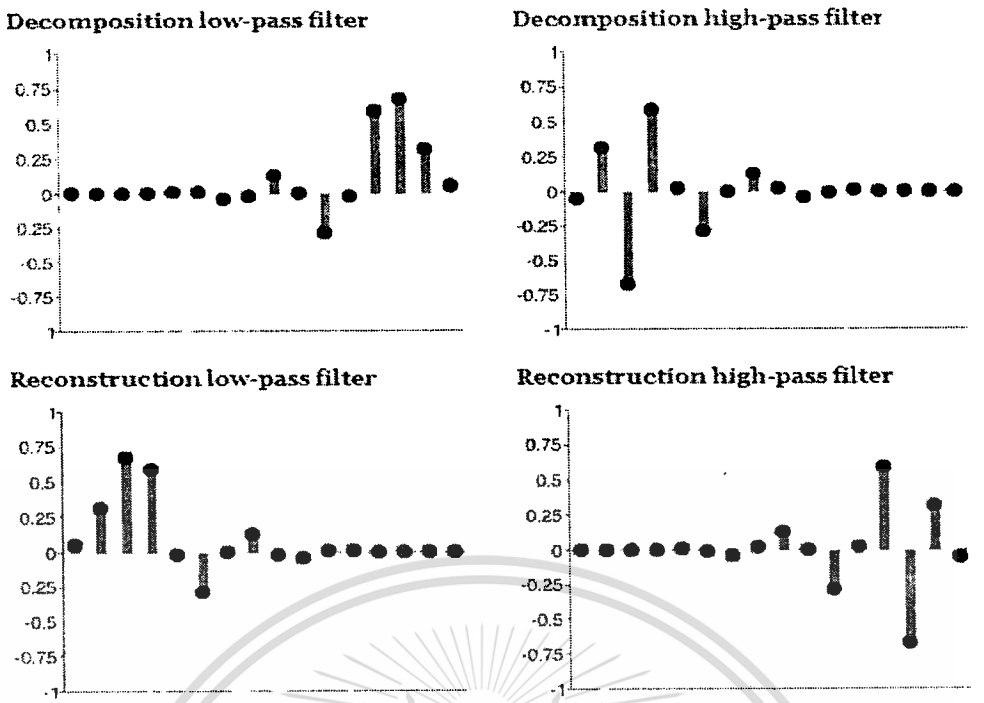
Figure 3.19 Daubechies wavelets: dbN

This family has the following properties: the ψ and φ support length is $2N$ -1. The number of zero moments of ψ is N; dbN wavelets are asymmetric (in particular for low values of N) except for the *Haar* wavelet; the regularity increases with order. When N becomes very large, ψ and φ belong to C^{μ_N} where $\mu N \approx 0.2$. This value μN is too pessimistic for relatively small orders, as it underestimates the regularity; and the analysis is orthogonal.

The wavelets of this family for the orders from 2 to 10 are presented in Figure 3.19 Moreover, for two of them (db4 and db8), we also find in Figure 3.20, apart from the wavelet, the scaling function and the four associated filters (two for decomposition, two for reconstruction).



a) Daubechies wavelets db4

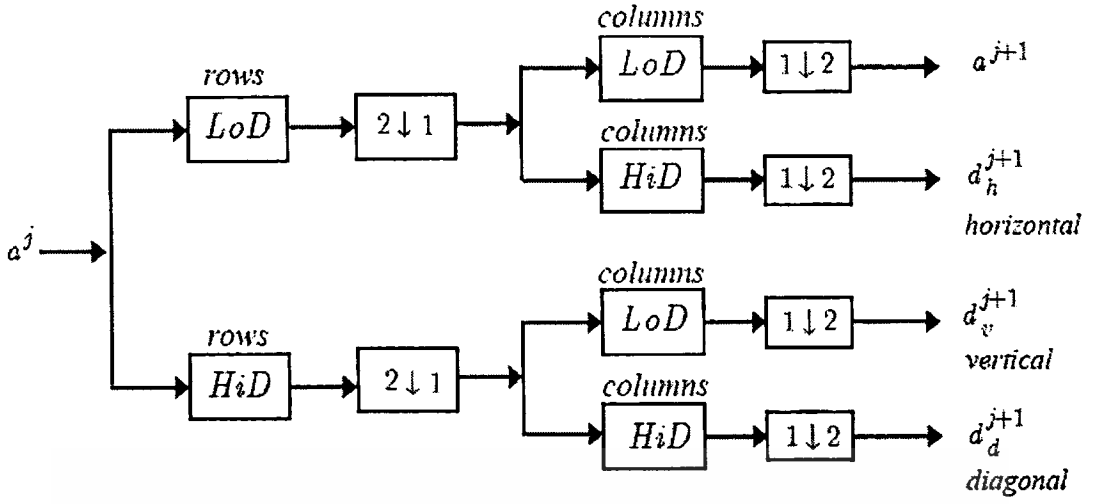


b) Daubechies wavelets db8

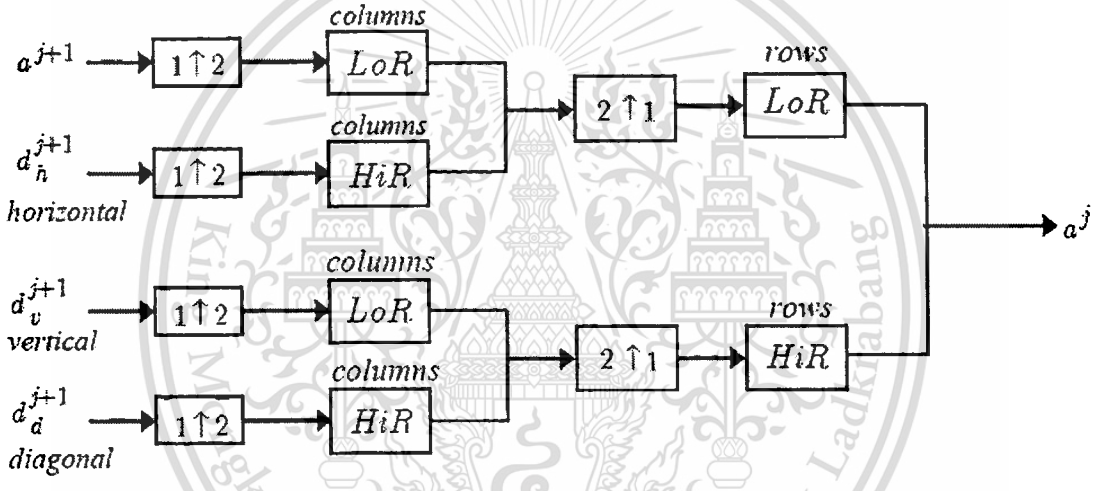
Figure 3.20 Two Daubechies wavelets.

3.10.3 From 1D to 2D

For the images, a similar algorithm is usable with wavelets and scaling functions obtained by the tensor product of one-dimensional wavelets. This type of transformation using a two-dimensional DWT leads to a decomposition of the approximation coefficients at the j level in four distinct components: the approximation and details according to three orientations, horizontal, vertical and diagonal, at the level $j + 1$. Calculations are simple: we filter the rows of a^j and decimate them; then, we filter and decimate the columns of the matrices obtained. Figure 3.21 a) presents the basic step of wavelet decomposition for images where F^{rows} (respectively $F^{columns}$) represents the convolution by F of rows (respectively columns) of the matrix, $2\downarrow1$ represents the decimation of the columns (conservation of even column index) and $1\downarrow2$ represents the decimation of the lines (conservation of even index lines). For the reconstruction, the basic step is given by Figure 3.21 b) where $2\uparrow1$ represents upsampling (zero insertion) of the columns and $1\uparrow2$ represents the upsampling of rows. It should be mentioned that for biorthogonal wavelets the same algorithms can be used, but the decomposition filters, on the one hand, and the reconstruction filters, on the other, are obtained from two distinct scaling functions associated with two multi-resolution analyses in duality.



a) The wavelets decomposition



b) The wavelets reconstruction

Figure 3.21 basic steps for the decomposition and reconstruction in wavelets for images.

3.10.4 Wavelets for the image

For images, a similar algorithm is possible for two-dimensional wavelets and scaling functions obtained from one-dimensional wavelets by tensorial product. Starting with a 1D multi-resolution analysis we note the associated approximation and detail spaces as V_j^{1D} and W_j^{1D} . For every level j the approximation space of the 2D multi-resolution analysis is obtained as a sum of four 1D tensor products:

$$V_{j-1}^{2D} = \overline{\left(V_j^{1D} \otimes V_j^{1D}\right)} \otimes \overline{\left(V_j^{1D} \otimes W_j^{1D}\right)} \otimes \overline{\left(W_j^{1D} \otimes V_j^{1D}\right)} \otimes \overline{\left(W_j^{1D} \otimes W_j^{1D}\right)} \quad (3.32)$$

This relation is also written:

$$V_{j-1}^{2D} = V_j^{2D} \otimes \left[W_j^{2D}\right]_h \otimes \left[W_j^{2D}\right]_v \otimes \left[W_j^{2D}\right]_d \tag{3.33}$$

Indeed, $V_{j-1}^{2D} = V_j^{1D} \otimes V_j^{1D}$ with $V_{j-1}^{2D} = V_j^{1D} \otimes W_j^{1D}$, which makes it possible to obtain the first expression for V_{J-1}^{2D}.

Whatever the method of 2D construction, whether using tensor products or not, we have a scaling function, as in 1D, and three wavelets instead of one. In the particular case of tensor construction, if φ and ψ indicate respectively the scaling function and the 1D wavelet, we have:

The scaling function: $\varphi^{2D}(x,y) = \varphi(x)\varphi(y)$, (3.34)

Three wavelets: $\psi_h^{2D} = \varphi(x)\psi(y)$, (3.35)

$$\psi_v^{2D} = \psi(x)\varphi(y), \tag{3.36}$$

$$\psi_d^{2D} = \psi(x)\psi(y) \tag{3.37}$$

The algorithms of decomposition and reconstruction of the 2D-DWT, with preceding tensor construction,

1) 2D wavelet decomposition

As in 1D, two types of objects are handled in 2D: approximation or detail coefficients corresponding to coordinates in the bases of spaces V_j^{2D} and W_j^{2D}; reconstructed approximations and details corresponding to projections on spaces V_j^{2D} and W_j^{2D}.

Here we are dealing with monochromatic images, i.e. in grayscale. An integer is associated with each pixel of the image. For a 2D signal noted X, we affect the values to the coordinates in V_0^{2D}. We break X up into a sum of orthogonal signals corresponding to different visualization or resolution scales. Thus, we have:

$$X = A_1 + D_1 = \cdots = A_j + D_j + \cdots + D_2 + D_1 \tag{3.38}$$

Decomposition along three directions of detail spaces implies that in 2D:

$$A_{j+1} = A_j + D_j = A_1 + \left[(D_h)_j + (D_v)_j + (D_d)_j\right] \tag{3.39}$$

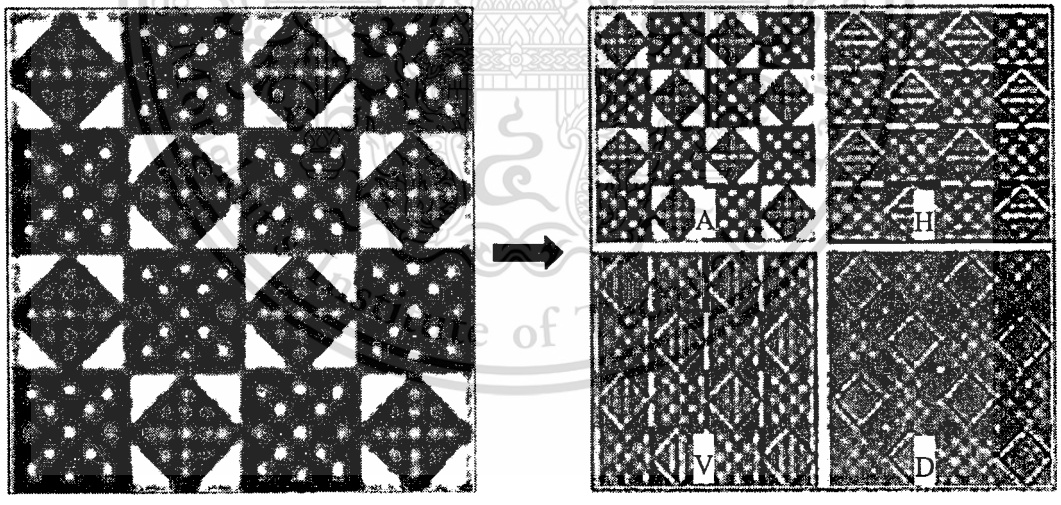
where D_h , D_v and D_d indicate respectively what is usually called horizontal, vertical and diagonal details.

An important point distinguishes the 1D from the 2D, it is the notion of "vision" attached to the 2D. In general, we process an image representing more than one simple function with two variables or a matrix because it implicitly contains the concept of "visual rendering".

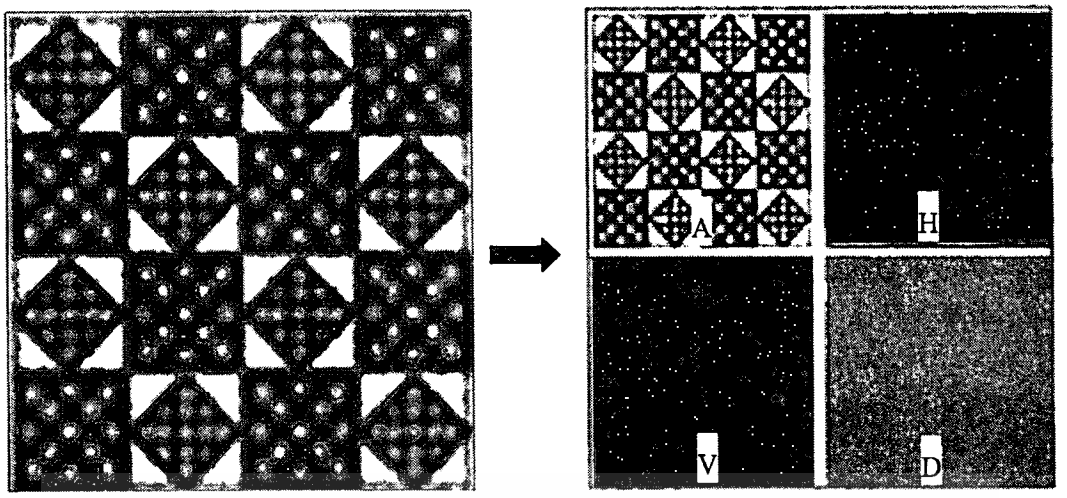
2) Approximation and detail coefficients

Horizontal, vertical and diagonal details: let our take an image extracted from an A. For which we carry out a level 1 analysis using the Haar and db4 wavelet. The initial image of size (256 × 256) presents marked geometric aspects. In particular, vertical and horizontal lines clearly stand out in the analyses. In Figure 3.22 on the left, the coefficients of approximation and the coefficients of detail of level 1, are represented as follows: *A*: approximation coefficients; and *H, V, D*: coefficients of horizontal, vertical and diagonal details respectively (see Figure 3.22)

We commonly use two types of representation for 2D wavelet decomposition. The first, already presented, highlights the proportions between the various components. The second representation highlights the arborescent aspect of wavelet decomposition.



a) With the wavelet Haar



b) With the wavelet db4

Figure 3.22 Decomposition at level one of the coefficients.

The coefficients obtained using the haar and db4 wavelet makes it possible to suitably distinguish the principal geometric aspects of the analyzed image. The use of another wavelet would make the direct reading of the coefficients more difficult, even impossible. To mitigate this drawback, the strategy that has to be adopted in order to interpret the results of the analysis is the same as in 1D: it is enough to consider the associated reconstructed signals.

3.11 Summary

In this chapter, the general techniques in image processing have been detailed. We, however, have included only those techniques that are quite necessary for fingerprint enhancement used in the next chapter. Filter techniques are common to many algorithms have been firstly proposed. Further, the filter techniques are taken into consideration for fingerprint image processing. They can improve images to be suitable for specific applications. The wavelet transforms is relatively new imaging tools that are being rapidly applied to a wide variety of image processing problems. This is because of their similarity to the Fourier transform.

Chapter 4

Fingerprint Image Enhancement

4.1 Fingerprint Image Enhancement

Fingerprint image enhancement is to make the fingerprint image clearer in order to ease further operations. Since the fingerprint images acquired from sensors or other media are not assured with perfect quality, those enhancement methods, for increasing the contrast between ridges and valleys and for connecting the false broken points of ridges due to insufficient amount of ink, are very useful for keep a higher accuracy to fingerprint identification. Fingerprints are well known for their uniqueness among individuals and this uniqueness is determined by the several ridge characteristics and their relationship. Out of more than one hundred ridge characteristics, two most prominent local ridge characteristics used for automatic fingerprint identification are ridge endings and ridge bifurcations (called minutiae). Where a ridge ends abruptly, that location is known as ridge ending. Where ridge forks or diverge into branch ridges, that location is known as ridge bifurcation. On another hand, singular points (core and delta point) are also very important in both classification and matching. Spectrum matching, for instance, requires center point or core point. Likewise, several classification algorithms use of ridge, and core and delta points.

Currently minutiae based fingerprint matching algorithm is being used in most widely used automatic fingerprint identification techniques. Hence, reliable extraction of minutiae is very important for good performance of identification technique. Extraction of minutiae depends significantly on quality of the fingerprint image. In case ridges and valleys are clearly visible and ridges can be easily detected and minutiae can be precisely located from a binary ridge map. However, in practice due to variation in impression conditions, ridge configuration, skin conditions, acquisition device and non-cooperative attitude of subjects, etc., a significant percentage of acquired fingerprint images is of poor quality. This can significantly affect the performance of fingerprint identification system.

To ensure the performance of feature extraction algorithms (singular points and minutiae) are robust with respect to the quality of fingerprint image, a fingerprint enhancement algorithm

which can improve the quality of fingerprint image, is of necessary. The goal of fingerprint enhancement algorithm is to improve the ridge structure in the input fingerprint image to facilitate the extraction of ridges and minutiae; on the other hand it should not create spurious ridge structures, as it may change the individuality of a fingerprint image.

There are appear many algorithms and techniques proposed and applied to fingerprint image enhancement: using Fourier transform [58, 60, 62], Gabor filters [3, 57, 59, 63, 73], Pyramid techniques [68, 69], Wavelet transform [66, 71, 72, 73, 74], and minutiae filtering, applied to binary [75] or gray-scale images [76]. The main of an enhancement algorithm is to improve the clarity of ridge structures of fingerprint images in recoverable regions and to remove the unrecoverable regions. This thesis, we propose method fingerprint image enhancement with directional filtering that involves two types similar of filters: Gabor filter and Second derivative of a Gaussian filter are also applied in the scheme of pyramid technique and directional wavelet transform that summarized our proposed as below:

In this [23, 77] approach involves the normalization of the fingerprint image so that it has a pre-specified mean and variance. Due to imperfections in the fingerprint image capture process such as non-uniform ink intensity or non-uniform contact with the fingerprint capture device, a fingerprint image may exhibit distorted levels of variation in grey-level values along the ridges and valleys. Thus, normalization is used to reduce the effect of these variations, which facilitates the subsequent image enhancement steps.

An orientation image is then calculated [3, 23, 57, 77], which is a matrix of direction vectors representing the ridge orientation at each location in the image. The widely employed gradient-based approach is used to calculate the gradient, which makes use of the fact that the orientation vector is orthogonal to the gradient. Firstly, the image is partitioned into square blocks and the gradient is calculated for every pixel, in the x and y directions. The orientation vector for each block can then be derived by performing an averaging operation on all the vectors orthogonal to the gradient pixels in the block. Due to the presence of noise and corrupted elements in the image, the ridge orientation may not always be correctly determined. Given that the ridge orientation varies slowly in a local neighborhood, the orientation image is then smoothed using a low-pass filter to reduce the effect of outliers.

Next step in the image enhancement process is the estimation of the ridge frequency image. The frequency image defines the local frequency of the ridges contained in the fingerprint.

Firstly, the image is divided into square blocks and an oriented window is calculated for each block. For each block, an x-signature signal is constructed using the ridges and valleys in the oriented window. The x-signature is the projection of all the grey level values in the oriented window along a direction orthogonal to the ridge orientation. Consequently, the projection forms a sinusoidal-shape wave in which the centre of a ridge maps itself as a local minimum in the projected wave. The distance between consecutive peaks in the x-signature can then be used to estimate the frequency of the ridges.

Fingerprint enhancement methods based on the Gabor filter have been widely used to facilitate various fingerprint applications such as fingerprint matching and fingerprint classification. Gabor filters are bandpass filters that have both frequency-selective and orientation-selective properties, which mean the filters can be effectively tuned to specific frequency and orientation values. One useful characteristic of fingerprints is that they are known to have well defined local ridge orientation and ridge frequency. Therefore, the enhancement algorithm takes advantage of this regularity of spatial structure by applying Gabor filters that are tuned to match the local ridge orientation and frequency.

The fingerprint enhancement methods based on the second derivative of Gaussian filter, the second derivative-based operation just as smoothing is a basic operation in image so is the ability to take one or more spatial derivatives of the image [78]. The basic problem is that, according to the mathematical definition of a derivative. The second derivative it is, of course, possible to compute higher order derivatives of functions of two variables in image and the second derivative of Gaussian filter is the straightforward extension of the Gaussian first derivative filter and can be applied independently in each dimension. The second derivative filter, on the other hands, depends very much on the orientation and ridge frequency. Because the local orientation changes very rapidly in the core point area, we almost can improve and make it accurate. Most of those we need of prior field smoothing similar to the filtering stage applied by Gabor filter. Hence, there have been many methods attempt to solve this problem.

The pyramid technique task of fingerprint enhancement is to counteract the aforesaid quality impairments and to reconstruct the actual fingerprint pattern as true to the original as possible. The latter part is especially noteworthy. There are many different filters that can be used for this purpose. In this study, we use of an image-scale pyramid for fingerprint image enhancement to improve the ridges and valleys. The pyramid technique decomposes the image

into a set of low pass filtered images. A continuous reduction of image resolution with respect to the original image can be achieved by smoothing the image with a Gaussian-smoothing kernel and the scale parameter corresponding to the standard deviation. Such successive smoothing of the original image gives a set of low-pass filtered images, which when stacked one on top of the other give rise to the Gaussian pyramid. A more detailed description of the Gaussian and Laplacian filter can be found in [68, 69, 79, 80].

The techniques of fingerprint enhancement applied directional wavelet, that wavelet transform has been widely used for image compression and fingerprint enhancement making possible the detection of redundant information. This wavelet property has been applied for fingerprint compression. Some works have also proposed the use of wavelets for fingerprint enhancement [66, 71, 72, 73, 74]. Other approaches have used the capability of the wavelet transform to distinguish the characteristics of the information inside images. From this point of view, several works have focused on determining a fingerprint feature vector extracted from the fingerprint wavelet transform.

Overall, it can be seen that most techniques for fingerprint image enhancement are based on directional filters that are tuned according to the local characteristics of fingerprint images. Our enhancement techniques employ the ridge orientation information for tuning of the filter. However, the orientation and ridge frequency information, it allows for accurate tuning of the Gabor filter parameters, which consequently leads to better enhancement results. Hence, we propose fingerprint enhancement with directional filtering that according to approach by Hong et al. [57] to perform fingerprint image enhancement.

Finally the performance of enhancement image is measured for its improvement by testing the success of core point identification where Poincare technique is used. The core point and delta point are ones of fingerprints' features that commonly used in fingerprint classification [81, 82, 83]. Classification process is of more importance when one wants to search a particular pattern in a large database. There are also many techniques used in searching for the core point; for instance, direction of curvature (DC), geometry region (GR), and Poincare method. Among these, Poincare technique can also detect a delta point. The Poincare technique accomplishes its task by checking the field orientation around the considering point. The success of point identification is solely depended on the correctness of the field orientation. In many cases, fields

are distorted by the discontinuity of the ridges. Smoothing the field is then helpful when one tries to eliminate the alias points created according to filed distortion.

In this chapter we propose to enhance a fingerprint image i.e., to estimate the ridge flow in low quality areas (wet, dry, scars etc). Hence, provides discussion on the method and implementation of a fingerprint image enhancement algorithm. First, I will review existing techniques in the field of fingerprint image enhancement. This will be followed by a description of the methods used to implement the enhancement algorithm. The results of the experiments involving each stage of the fingerprint enhancement algorithm will then be presented and discussed.

4.2 Fingerprint Enhancement Process

In this thesis, we propose a method to enhance the quality of fingerprint images by using directional filtering. Those techniques mainly consist of two filters, i.e., Gabor filtering and the second derivatives of Gaussian filter. These filters are also used in scheme of pyramid technique and directional wavelet transform. To give well understanding of the technique used, this chapter is devoted to several sub-processes. In this section we review fingerprint enhancement process. It can be summarized as followings: normalization, field estimation field smoothing, ridge frequency estimation and directional filtering. These are details in this section. In section 4.3, we review directional filtering steps as follows fingerprint enhancement using: Gabor filter, the second derivative filter, pyramid technique and wavelet transform. Finally in section 4.4, we review an algorithm used for core point detection, a Poincare technique.

4.2.1 Normalization

The normalization is basically employed to standardize the intensity values in an image by adjusting the range of grey-level values so that it lies within a desired range of values. Let, I(x, y) denote the gray-level value at pixel (i, j), M and VAR denote the estimated mean and variance of image I, respectively, and N(i, j) denote the normalized gray-level value at pixel (x, y). The normalized image is defined as:

$$N(x,y)=\begin{cases} M_0+\sqrt{\dfrac{VAR_0(I(x,y)-M)^2}{VAR}} & if\ I(x,y) \\ M_0-\sqrt{\dfrac{VAR_0(I(x,y)-M)^2}{VAR}} & otherwise \end{cases} \tag{4.1}$$

Where, M_0 and VAR_0 are the desired mean and variance values, respectively. In this work, we set values mean M_0 = 0.5 and variance VAR_0 = 1. The mean and variant of a gray-level fingerprint image with the dimension of $M \times N$ pixels are defined respectively as:

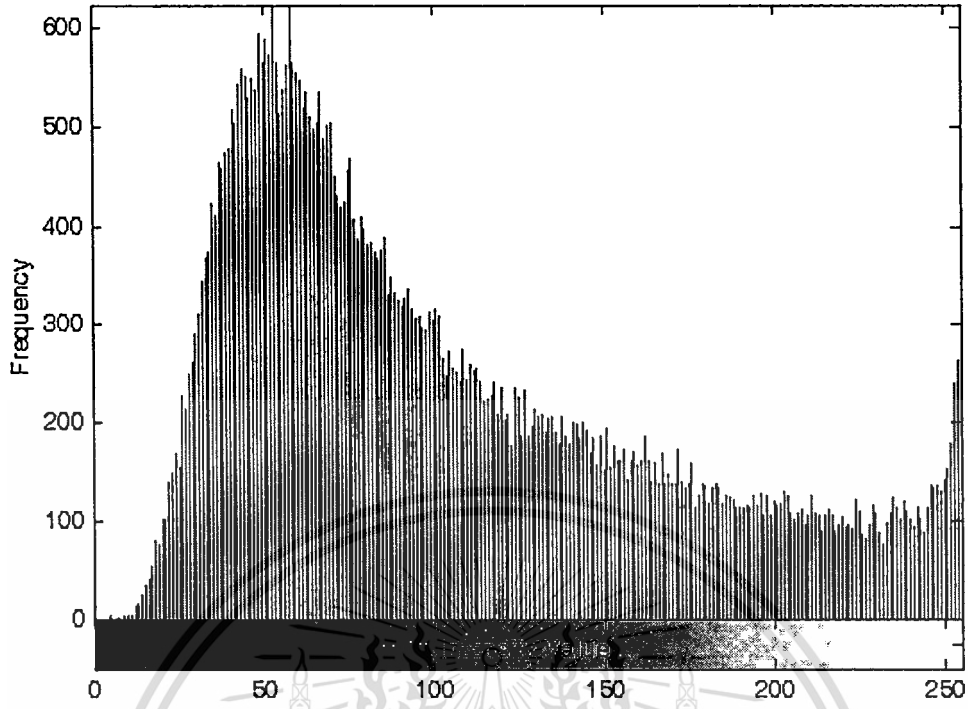
$$M(I) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x,y) \tag{4.2}$$

and

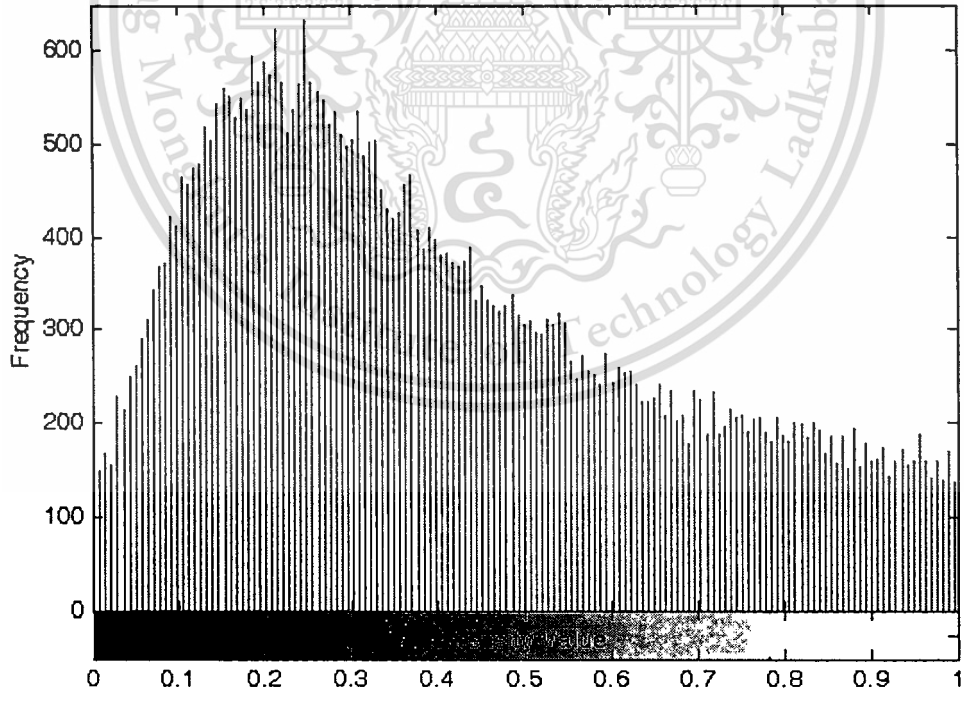
$$VAR(I) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left(I(x,y) \_ M(x,y)\right)^2 \tag{4.3}$$

Where $I(x,y)$, represents the intensity of the pixel at x^{th} row and y^{th} column. The main purpose of the normalization operation is to reduce the variations of gray-level values along the ridges and valleys. The operation is pixel-wise and does not change the clarity of the ridges and valleys as Figure 4.1 shown example of histogram of the normalization.

a) Original image b) Normalization



c) Histogram of original image



d) Histogram of normalized image

Figure 4.1 Example of histogram of the normalization.

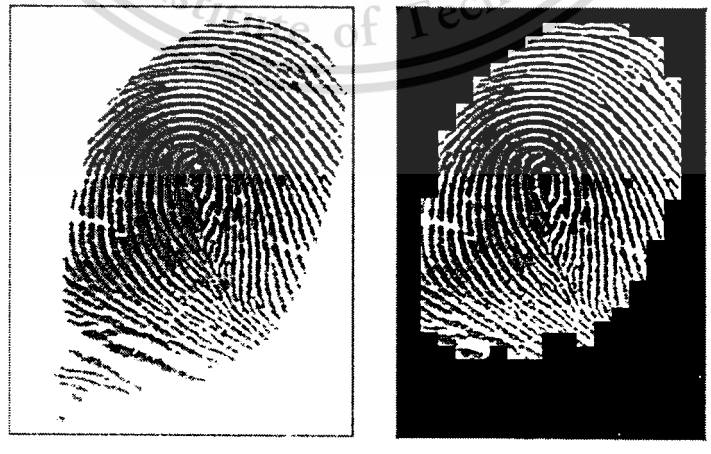
4.2.2 Segmentation

The segmentation is the process of separating the foreground regions in the image from the background regions. The foreground regions correspond to the clear fingerprint area containing the ridges and valleys, which is the area of interest. The background corresponds to the regions outside the borders of the fingerprint area, which do not contain any valid fingerprint information. When minutiae extraction algorithms are applied to the background regions of an image, it results in the extraction of noisy and false minutiae. Thus, segmentation is employed to discard these background regions, which facilitates the reliable extraction of minutiae.

In a fingerprint image, the background regions generally exhibit a very low grey-scale variance value, whereas the foreground regions have a very high variance. Hence, a method based on variance thresholding can be used to perform the segmentation. Firstly, the image is divided into blocks and the grey-scale variance is calculated for each block in the image. If the variance is less than the global threshold, then the block is assigned to be a background region; otherwise, it is assigned to be part of the foreground. The grey-level variance for a block of size $W \times W$ is defined as:

$$V(k) = \frac{1}{W^2} \sum_{x=0}^{W-1} \sum_{y=0}^{W-1} (I(x,y) - M(k))^2 \tag{4.4}$$

Where $V(k)$ is the variance for block k, $I(x,y)$ is the grey-level value at pixel $(x; y)$, and $M(k)$ is the mean grey-level value for the block k and Figure 4.2 illustrated Example of segmentation.



a) Original image b) Segmentation

Figure 4.2 Example of fingerprint image segmentation.

4.2.3 Orientation field estimation

The ridge orientation (orientation field) at a pixel (x, y) is the angle θ_{xy} that the fingerprint ridges, crossing through an arbitrary small neighborhood centered at (x, y), form with the horizontal axis. Because fingerprint ridges are not directed, θ_{xy} is an unoriented direction lying in [0 -180°]. In the rest of the thesis we use the term orientation to denote an unoriented direction in [0 -180°], and the term direction to indicate an oriented direction in [0 - 360°].

Instead of computing ridge orientation field at each pixel, most of the fingerprint processing and feature extraction methods estimate the ridge orientation at discrete positions (this reduces computational efforts and still allows estimates at other pixels to be obtained through interpolation). The fingerprint orientation image (also called directional image), first introduced by [84] is a matrix whose elements encode the local orientation of the fingerprint ridges. Each element θ_{ij}, corresponding to the node (i, j) of a square-meshed grid located over the pixel (x_i, y_j), denotes the average orientation of the fingerprint ridges in a neighborhood of (x_i, y_j) (see Figure 4.3). Each element denotes the orientation field of the fingerprint ridges; the element length is proportional to its reliability. An additional value r_{ij} is often associated with each element θ_{ij} to denote the reliability (or consistency) of the orientation. The value r_{ij} is low for noisy and seriously corrupted regions and high for good quality regions in the fingerprint image.

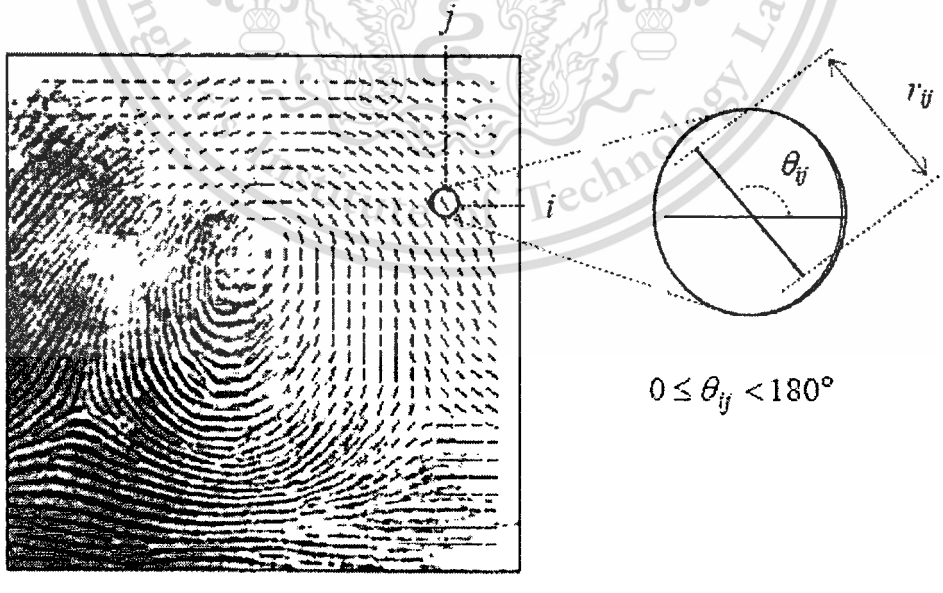


Figure 4.3 A fingerprint image faded into the corresponding orientation image computed over a square-meshed grid of size 16×16.

Gradient-based approaches: the simplest and most natural approach for extracting local ridge orientation is based on computation of gradients in the fingerprint image. The gradient $\partial(x, y)$ at point (x, y) of I, is a two dimensional vector $[\partial x(x, y), \partial y(x, y)]$, where ∂x and ∂y components are the derivatives of I at (x, y) with respect to the x and y directions, respectively. It is well known that the gradient phase angle denotes the direction of the maximum intensity change. Therefore, the direction θ of a hypothetical edge crosses the region deterred at (x, y) is orthogonal to the gradient phase angle at (x, y). This method, although simple and efficient, has some drawbacks. First using the classical Sobel convolution mask [6, 57, 77] to determine ∂x and ∂y components of the gradient, and computing θ as the arctangent of $\partial y/\partial x$ ratio, presets problem due to non-linearity and discontinuity around 90°. Second, a single orientation estimate reflects the ridge and valley orientation at too fine a scale and is generally very sensitive to the noise in the fingerprint image; On the other hand, simply averaging gradient estimate is not meaningful due to circularity of angles: the average orientation between 5° and 175° is not 90° (as an arithmetic average suggest) but 0°. Furthermore, the concept of average orientation is not always well define; consider the two orthogonal orientation 0° and 90° ; is the correct average orientation 45° and 135°. In [85] proposed a simple but smooth solution to the above problem, which allows local gradient estimate is encode by vector:

$$d = [r\cos 2\theta, r\sin 2\theta] \tag{4.5}$$

Where 2θ is used in place of θ to discount the circularity of angles and r is proportion to the orientation estimate strength (e.g., the square norm of gradient: $\partial_x^2 + \partial_y^2$). Averaging angles in local $n \times n$ window W to obtain a more robust estimate $\bar{d}$, can be performed by separately averaging the two (x and y) components:

$$\bar{d} = \left[\frac{1}{n^2} \sum_W r\cos 2\theta, \frac{1}{n^2} \sum_W r\sin 2\theta, \right] \tag{4.6}$$

Computing the average between two orthogonal orientations with equation (4.6) involves summing to vectors facing each other. and therefore the length of the resulting vector is zero. This indicates that vector is meaningless, independent of its orientation.

$$\theta_{ij} = 90° + \frac{1}{2} a\tan 2\left(2G_{xy}, G_{xx} - G_{yy}\right) \tag{4.7}$$

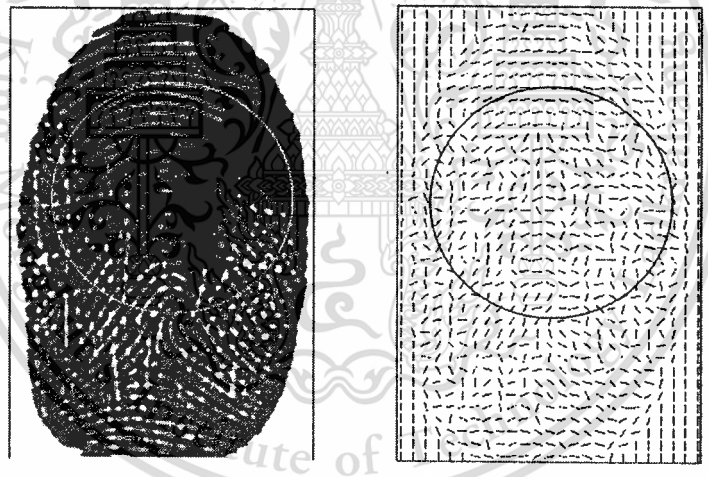
Where,

$$G_{xy} = \sum_{h=-8}^{8} \sum_{k=-8}^{8} \partial_x(x_i + h, y_j + k)\partial_y(x_i + h, y_j + k) \tag{4.8}$$

$$G_{xx} = \sum_{h=-8}^{8} \sum_{k=-8}^{8} \partial_x(x_i + h, y_j + k)^2 \tag{4.9}$$

$$G_{yy} = \sum_{h=-8}^{8} \sum_{k=-8}^{8} \partial_y(x_i + h, y_j + k)^2 \tag{4.10}$$

where ∂x and ∂y are the x- and y-gradient components computed through 3x3 Sobel masks (see [57, 77]), and ***atan2(y, x)*** calculates the arctangent of the two variables y and x: it is similar to calculating the arctangent of y/x, except that the signs of both arguments are used to determine the quadrant of the result. An example of local orientation image computed with equation (4.7) is shown in Figure 4.4 b).

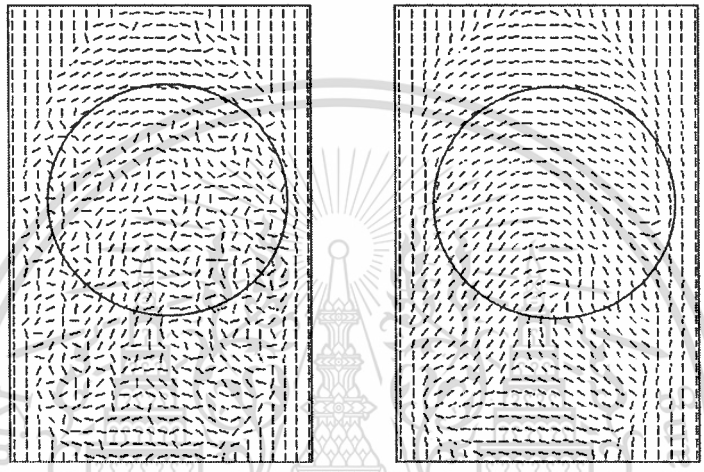


a) Original image b) Field estimation

Figure 4.4 Orientation field estimation is computed with equation (4.7).

The orientation field smoothing computed from poor quality fingerprints may contain several unreliable elements due to creases, local cluttered noise. In this situation, a field smoothing can be very useful in enhancing. This can be done by (re)converting the angles in orientation vectors **d** (equation (4.5)) and by averaging them through equation (4.6). Figure 4.5 shows an example of orientation image smoothing. However, such a simple averaging has some limitations (Figure 4.5 b)): It is ineffective when the incorrect orientations dominate the correct

ones; tends to smooth out high curvature values, especially in singular point regions; and tends to slightly shift the loop singularities. Figure 4.5 a) field estimation of in a fingerprint through the gradient-based approach corresponding to equation (4.7), in the noisy regions the estimation is unreliable; b) increase window of filter smoothing are applied, resulting in a more consistent representation; it is worth noting that while the smoothing recovered the correct orientation at several places (e.g., inside the solid circle), it altered the average orientation inside the region denoted by the circle where incorrect orientations were dominating the correct one.



a) Field estimation b) Field smoothing

Figure 4.5 Example of orientation field smoothing.

4.2.4 Ridge frequency estimation

The local ridge frequency (or density) f_{xy} at point (x, y) is the number of ridges per unit length along a hypothetical segment centered at (x, y) and orthogonal to the orientation field θ_{xy}. A frequency image, analogous to the orientation image, can be defined if the frequency is estimated at discrete positions and arranged into a matrix. The ridge frequency estimation varies across different fingers, and may also noticeably vary across different regions of the same fingerprint. The next step is to project the grey-level values of all the pixels located inside each block along a direction orthogonal to the local ridge orientation. This projection forms an almost sinusoidal-shape wave with the local minimum points corresponding to the ridges in the fingerprint. An example of a projected waveform is shown in Figure 4.6.

I have referred the original frequency estimation stage used by [57, 77] to include an additional projection smooth step prior to computing the ridge spacing. This involves smoothing

the projected waveform using a Gaussian lowpass filter of size to reduce the effect of noise in the projection. The ridge spacing $T(x,y)$ is then computed by counting the median number of pixels between consecutive minima points in the projected waveform. Hence, the ridge frequency $f(x,y)$ for a block center at pixel (x_i, y_j) is defined as $f_{xy} = 1/T(x,y)$. Given that the fingerprint is scanned at a fixed resolution, then ideally the ridge frequency values should lie within a certain range. However, there are cases where a valid frequency value cannot be reliably obtained from the projection. Examples are when no consecutive peaks can be detected from the projection, and also when minutiae points appear in the block. For the blocks where minutiae points appear, the projected waveform does not produce a well-defined sinusoidal shape wave, which can lead to an inaccurate estimation of the ridge frequency. Thus, the out of range frequency values are interpolated using values from neighboring blocks that have a well-defined frequency.

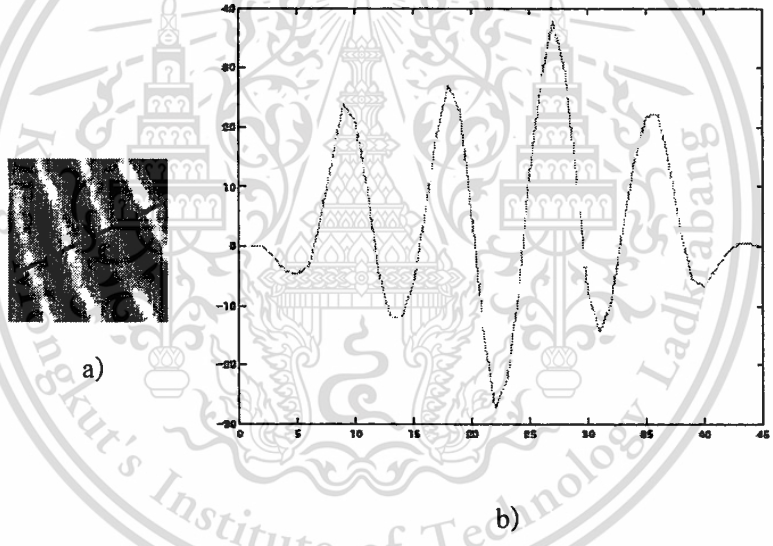


Figure 4.6 The projection of the intensity values of the pixels along a direction orthogonal to the local ridge orientation.

The orientation field estimation, the ridge frequency estimation is important parameter used in the construction of the directional filtering. Note that for the ridge frequency values will be presented in terms of ridge wavelength for easier interpretation of the results. For example, if the ridge spatial frequency value is 1/10 pixels. We modify the original wavelength estimation stage used by [77] to include smoothing of the projected waveform prior to computing the ridge wavelength (see Figure 4.6), a) A 32 × 32 block from a fingerprint image, b) the projected waveform of the block .The results for well-defined images (see Figure 4.7) show that the

majority of the estimated wavelength values for each 32 x 32 block match up accurately with the actual wavelength value of 10.

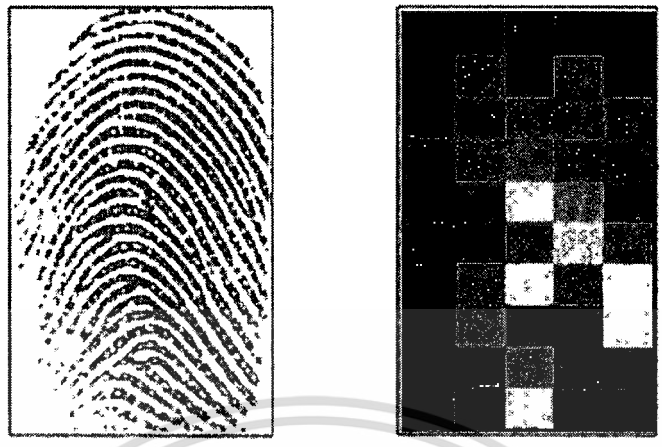


Figure 4.7 The estimated ridge wavelength (ridge frequency) for fingerprint images of wavelength.

4.2.5 Directional filtering

Directional filtering base on the estimated field is applied. The gray scale image is enhanced eventually. Two similar filters will be given in more details in next section. The said filters are extended to corporate with wavelet and pyramid technique.

4.3 Directional Filtering

Directional filtering is of interest when one wants to highlight the image pattern or texture holding particular direction or orientation. Gabor filter is linear filter used in image processing. Its impulse response in defined by a harmonic function multiplied by a Gaussian function. Similarly, second derivative of Gaussian function modified by multiplying with a cosine function can hold the feature of directional filtering.

4.3.1 Fingerprint enhancement using Gabor filter

With an original fingerprint image there are noises, and may be low quality such as dry, wet, damped, scars, smudges and so on. In most case, the degradation occurs in part. Without any attempts, it is hard to identify the core point and minutiae of such an image with those degradations. However, at this state noise could be removed by filtering techniques. Among those, Gabor filter is a promising one.

An effective method based on Gabor filters. Gabor filters have both frequency-selective and orientation-selective properties and have optimal joint resolution in both spatial and frequency domains. As shown in Figure 4.8, a Gabor filter is defined by a sinusoidal plane wave (the second term of equation (4.11)) tapered by a Gaussian (the first term in equation (11)). Figure 4.9 Appearance of Gabor filter, only 0° and 90° oriented filters (mask size = 32x 32, $f = 0.11$, $\sigma_x = 4$, $\sigma_y = 4$). The even symmetric two-dimensional Gabor filter has the following form.

$$G(x, y : \theta, f) = \exp\left[-\frac{1}{2}\left(\frac{x_\theta^2 + y_\theta^2}{\sigma_x^2 + \sigma_y^2}\right)\right]\cos(2\pi f x_\theta) \tag{4.11}$$

where θ is the orientation of the filter, and (x_θ, y_θ) are the coordinates of (x, y) after a clockwise rotation of the Cartesian axes by an angle of $(90°-\theta)$.

$$\begin{bmatrix} x_\theta \\ y_\theta \end{bmatrix} = \begin{bmatrix} \cos(90° - \theta) & \sin(90° - \theta) \\ -\sin(90° - \theta) & \cos(90° - \theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \sin\theta & \cos\theta \\ -\cos\theta & \sin\theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{4.12}$$

In the above expressions, f is the frequency of a sinusoidal plane wave, and σ_x and σ_y are the standard deviations of the Gaussian envelope along the x- and y-axes, respectively. Figure 4.8 defined by the parameters $\theta = 135°$, $f = 1/5$ and $\sigma_x = \sigma_y = 3$.

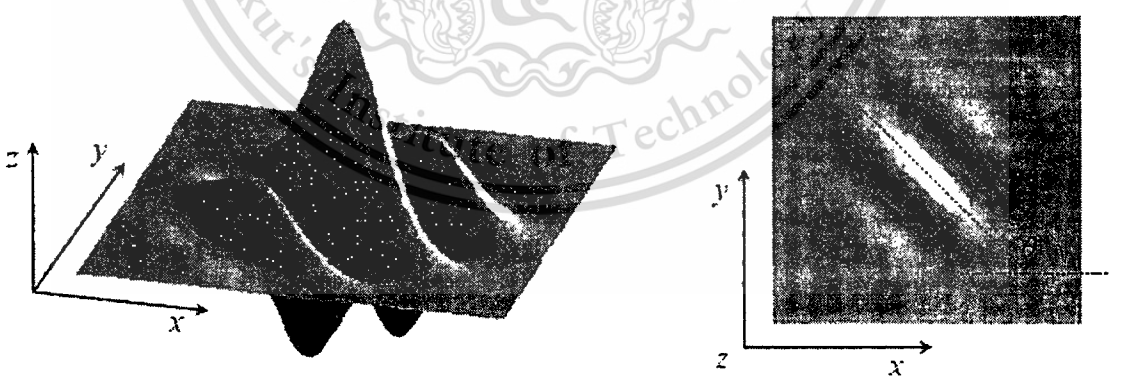
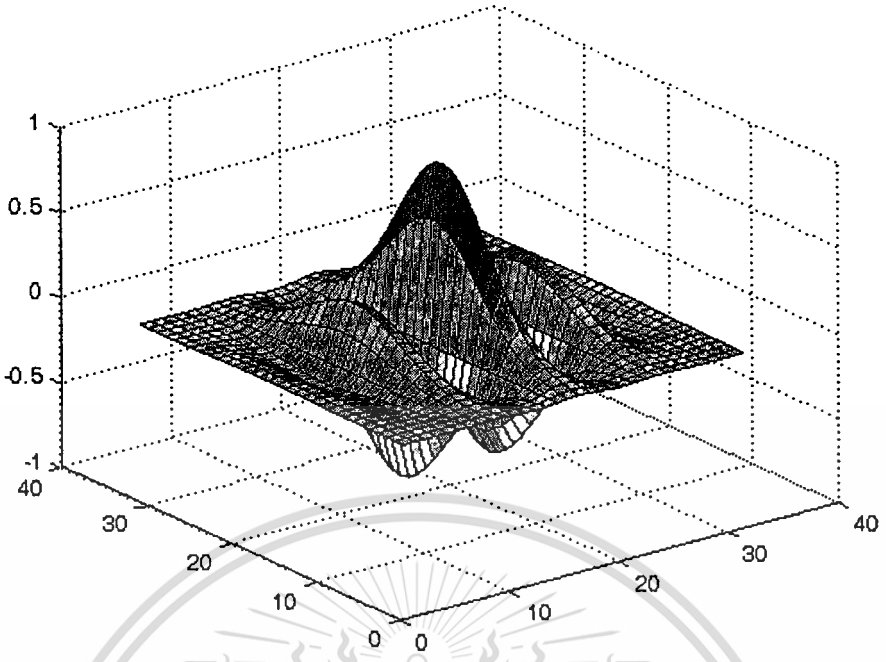
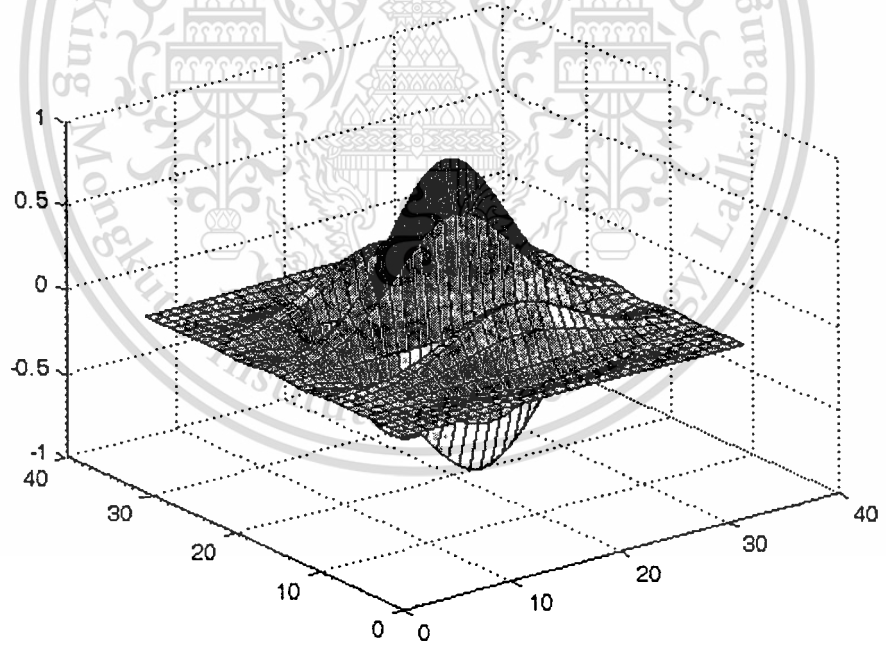


Figure 4.8 Graphical appearance of the Gabor filter.



a) $\theta = 0°$ orientation



b) $\theta = 90°$ orientation

Figure 4.9 Appearance of Gabor filters different of oriented filters.

To apply Gabor filters to an image, the four parameters $(\theta, f, \sigma_x, \sigma_y)$ must be specified. Obviously, the frequency of the filter is completely determined by the local ridge frequency and the orientation is determined by the local ridge orientation. The selection of the values σ_x and σ_y involves a tradeoff. The larger the values, the more robust the filters are to the noise in the fingerprint image, but they are also more likely to create spurious ridges and valleys. On the other hand, the smaller the values, the less likely the filters are to introduce spurious ridges and valleys but then they will be less effective in removing the noise. In fact, from the Modulation Transfer Function (MTF) of the Gabor filter, it can be shown that increasing σ_x and σ_y decreases the bandwidth of the filter and vice versa. Based on empirical data, [57] set $\sigma_x = \sigma_y = 4$.

To make the enhancement faster, instead of computing the best-suited contextual filter for each pixel "on the fly," a set $\{g_{ij}(x,y) \mid i = 1 \ldots n, j = 1 \ldots n\}$ of filters are a prior created and stored, where n is the number of discrete orientations $\{\theta_i \mid i = 1 \ldots n\}$ and n the number of discrete frequencies $\{\{f_j \mid j = 1 \ldots n\}$. Then each pixel (x, y) of the image is convolved, in the spatial domain, with the filter $g_{ij}(x,y)$ such that θ_i is the discretized orientation closest to θ_{xy} and f_j is the discretized frequency closest to f_{xy}. Figure 4.10 shows an example of the filter set for $n_o = 8$ and $n_f = 3$ Gabor filters where $\sigma_x = \sigma_y = 4$.

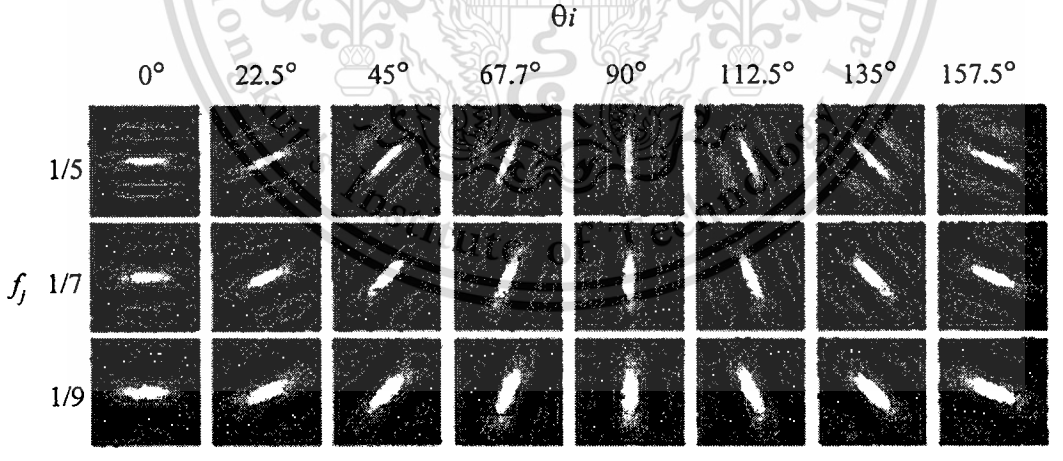


Figure 4.10 A graphical representation of a bank of 24.

A drawback of using fixed values is that it forces the bandwidth to be constant, which does not take into account the variation that may occur in the values of the ridge frequency. For example, if a filter with a constant bandwidth is applied to a fingerprint image that exhibits significant variation in the frequency values, it could lead to non-uniform enhancement or other enhancement artifacts. Thus, rather than using fixed values, I have chosen the values of σ_x and σ_y to be a function of the ridge frequency parameter, which is defined as $\sigma_x = k_x f(x,y)$ and $\sigma_x = k_y f(x,y)$, where f is the ridge frequency image, k_x is a constant variable for σ_x, and k_y is a constant variable for σ_y. This allows a more adaptable approach to be used, as the values of σ_x and σ_y can now be specified adaptively according to the local ridge frequency of the fingerprint image. Figure 4.11 shows the application of Gabor-based contextual filtering on poor quality images first row a) original and second row b) the ridge enhanced recoverable regions are clearly.

a) Original images

b) Ridges enhancement

Figure 4.11 Examples of fingerprint enhancement is applied Gabor filter (GBF).

4.3.2 Fingerprint enhancement using the second derivative of Gaussian filter

This technique is an improvement over the previous method in the sense that in this technique the frequency of the image is also taken into account as compared to previous method where in the frequency was taken to be constant throughout. This technique is similar previous method to that of Gabor filter.

The Second derivatives of Gaussian filter: The derivative-based operation just as smoothing is a basic operation in image so is the ability to take one or more spatial derivatives of the image. The basic problem is that, according to the mathematical definition of a derivative. The second derivative it is, of course, possible to compute higher order derivatives of functions of two variables in image and the second derivative of Gaussian filter is the straightforward extension of the Gaussian first derivative filter and can be applied independently in each dimension. In normal fingerprint image, the second derivative is compute higher-order derivatives of functions of two dimensions. The sinusoidal-shaped waves of ridges and valleys vary slowly in a local constant orientation. Therefore, a band-pass filter that is tuned to the corresponding frequency and orientation can efficiently remove the undesired noise while preserving the true ridge and valley structures. Note the parameters and values of the second derivative of Gaussian filter are used, it's similar to the Gabor filter (see above section 4.3.1).

Once, 2-D circular Gaussian filter is mostly adopted as an image preprocessing step for image smoothing and denoising, while the 2-D elliptical Gaussian filter can be used for enhancement of lines. Figure 4.12 is appearance of the elliptical Gaussian filter and is formulated as follows:

$$g(x,y) = \exp\left[-\left(\frac{x}{\sigma_x}\right)^2 - \left(\frac{y}{\sigma_y}\right)^2\right] \tag{4.13}$$

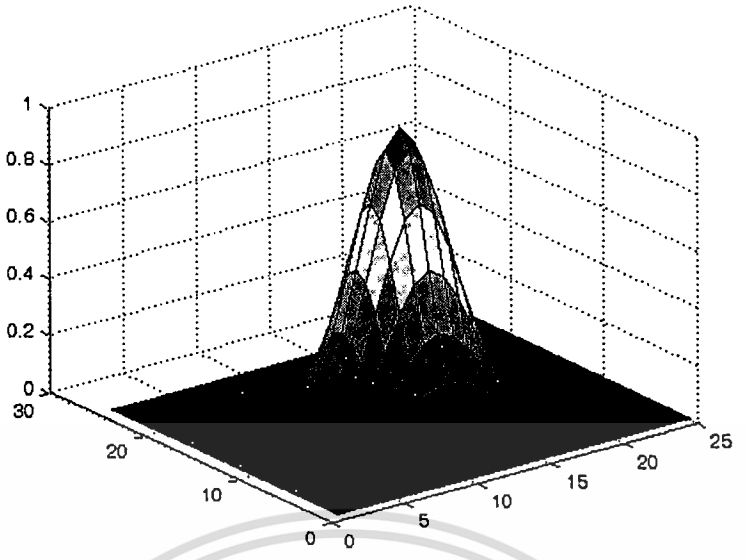


Figure 4.12 Appearance of Gaussian filter

The 2-D Gaussian filter is mostly adopted as an image preprocessing step for image smoothing and denoising. The second derivative of Gaussian filter given in equation (4.13) is the straightforward extension of the Gaussian first derivative filter. This can be applied independently to each dimension.

$$sdg(x,y:\theta)=\frac{(x_\theta^2-\sigma_\theta^2)(y_\theta^2-\sigma_\theta^2)}{2\pi\sigma_\theta^{10}}\exp\left(-\frac{x_\theta^2+y_\theta^2}{\sigma_\theta^2}\right) \tag{4.14}$$

This second derivative can enhance the ridge and suppress the valley for certain degree. To enhance the effectiveness of the filter we can modify equation (4.14) slightly by co-operating the cosine function (or plan wave) as follow:

$$sdg(x,y:\theta,f)=\frac{(x_\theta^2-\sigma_\theta^2)(y_\theta^2-\sigma_\theta^2)}{2\pi\sigma_\theta^{10}}\exp\left(-\frac{x_\theta^2+y_\theta^2}{\sigma_\theta^2}\right)\cos(2\pi f x_\theta) \tag{4.15}$$

Where, θ is the orientation of the second derivative filter, f is the frequency of a sinusoidal plane wave, σ_θ ($\sigma_\theta=4$) is the standard deviations of the Gaussian envelope. x_θ and y_θ define the x and y axes of the filter coordinate frame. Figure 4.13 is appearance of the second derivative of the Gaussian filter (similar to the appearance Gabor filter in Figure 4.9 a)). Figure 4.14 shown example of fingerprint enhancement applied the Second derivative of Gaussian filter, first row a) original and second row b) the ridge enhanced recoverable regions are improves and clearly.

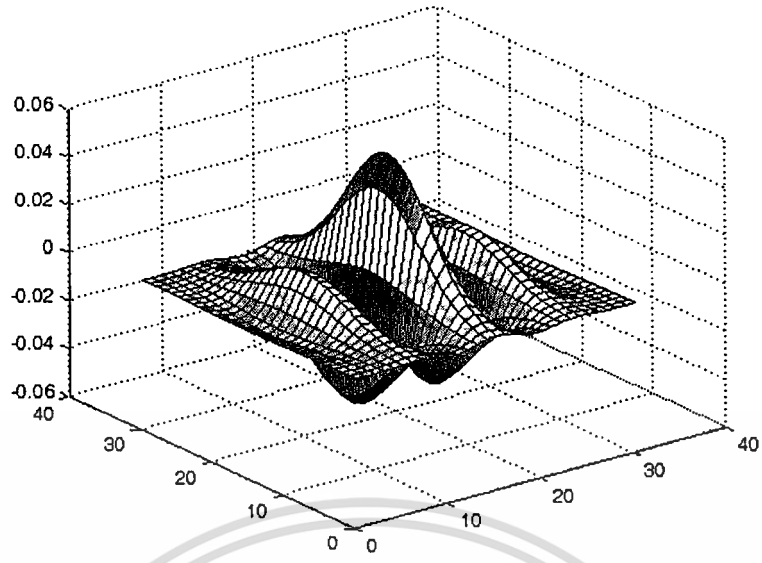


Figure 4.13 Appearance of the second derivative of a Gaussian filter

a) Original images

b) Ridges enhancement

Figure 4.14 Examples of fingerprint enhancement is applied the Second derivative of Gaussian filter (SDGF).

4.3.3 Fingerprint enhancement using pyramid technique

Many acquired fingerprint images may accompany noises, and also may be are of low quality such as dry, wet, damped, scars, smudges and so on. In most case, the degradation occurs in part. Without any attempts, it is hard to identify the core point and minutiae of such images with those degradations. Several types of filter can be applied for the purpose of noise removal. However, at this state we are considering pyramid enhancement that shown in Figure 4.15 data flow diagram fingerprint enhancement with pyramid decomposition and reconstruction.

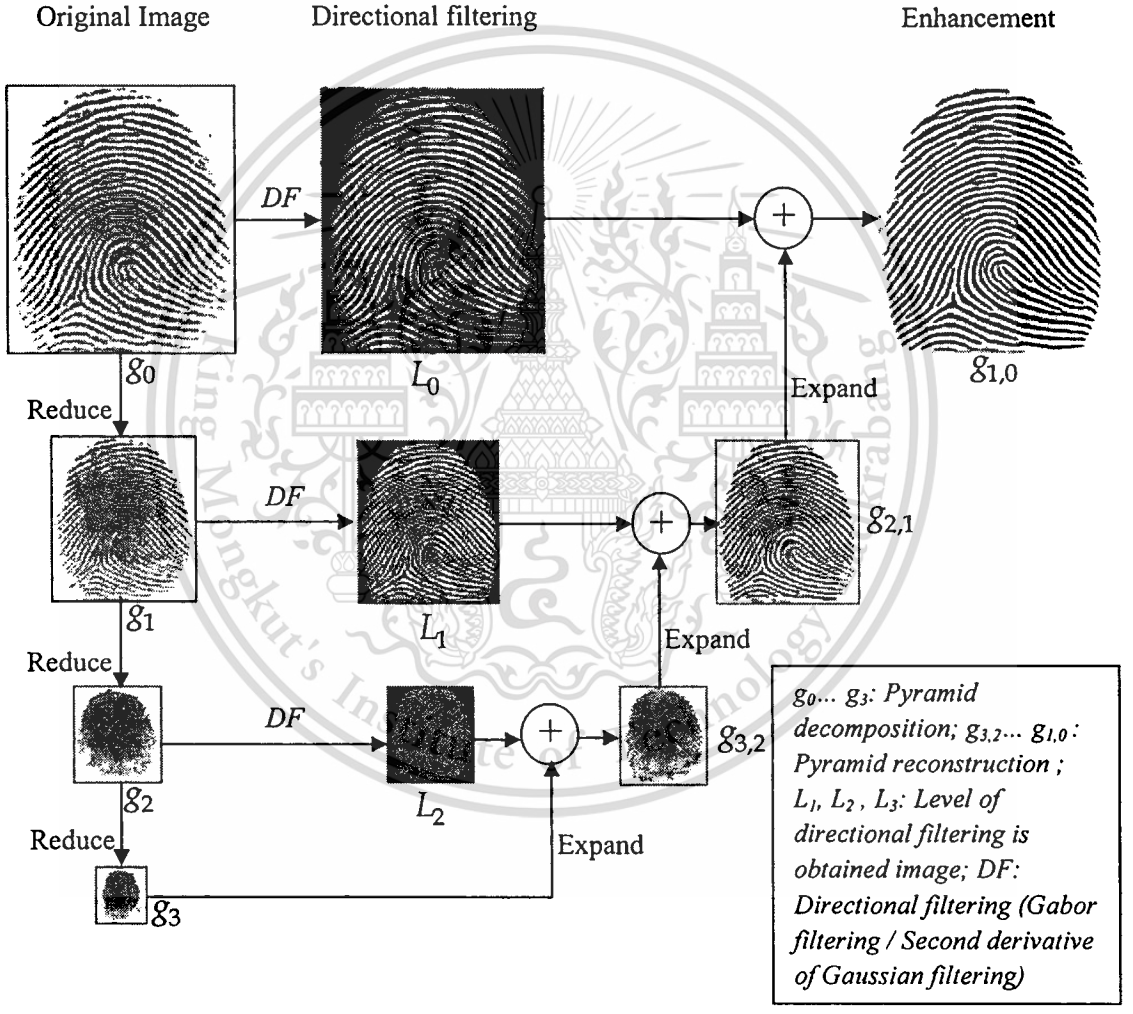


Figure 4.15 Data flow diagram fingerprint enhancement using pyramid technique

1) Pyramid decomposition

In this the construction of the pyramid technique (Gaussian pyramid) is a low pass filter and applies from the normalized input fingerprint image $N(x,y)$ the original image g_0, where $g_0 = N(x,y))$ to obtain an image g_1. We say that g_1 is a "reduced" version of g_0 in that both resolution and sample density are decreased. In a similar way we form g_2 as a reduced version of g_1, and last level form g_3 as a reduced version of g_2. Filtering is performed by a procedure equivalent to convolution with one of a family of local, symmetric weighting functions. An important member of this family resembles the Gaussian probability distribution, so the sequence of images g_0, $g_1 ... g_n$ is called the Gaussian pyramid (set of low pass filtered images). At each level, a directional filter (Gabor filter/Second derivative of Gaussian filter) is applied and the image L_0 , L_1 , L_3 is obtained. There is no need to apply the directional filtering to the last level (see Figure 4.15).

Gaussian pyramid generation: Suppose the image is represented initially by g_0 which contains columns and rows of pixels. Each pixel represents the light intensity at the corresponding image point. This image becomes the bottom or zero level of the Gaussian pyramid. Pyramid level 1 contains image g_1, which is a reduced or low-pass filtered version of g_0. Each value within level 1 is computed as a weighted average of values in level 0 within a 5-by-5 window. Each value within level 2, representing g_2, is then obtained from values within level 1 by applying the same pattern of weights. A graphical representation of this process in one dimension is given in Figue4.16 The size of the weighting function is not critical. We have selected the 5-by-5 pattern because it provides adequate filtering at low computational cost.

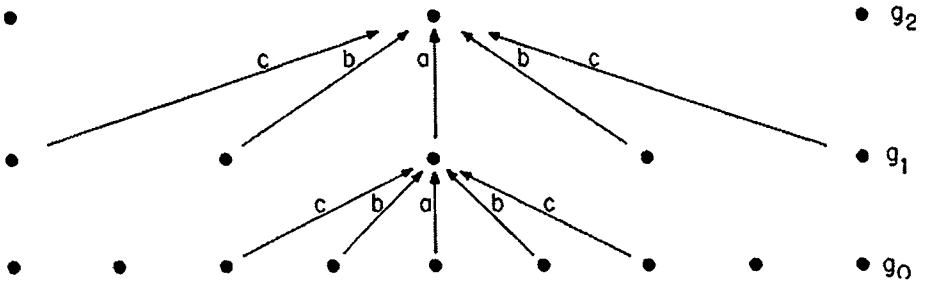
The level-to-level averaging process is performed by the function REDUCE.

$$g_l = REDUCE(g_{l-1}) \tag{4.16}$$

which means, for levels $-2 < l < 2$

$$g_l(i,j) = \sum_{m=-2}^{2} \sum_{n=-2}^{2} w(m,n) g_{l-1}(2i+m, 2j+n) \tag{4.17}$$

Note in Figure 4.16 shown the density of nodes is reduced by half in one dimension or by a fourth in two dimensions from level to level.



g_0 = IMAGE

$$g_l = REDUCE(g_{l-1})$$

Figure 4.16 A one-dimensional graphic representation of the process which generates a Gaussian pyramid each row of dots represents nodes within a level of the pyramid.

The value of each node in the zero level is just the gray level of a corresponding image pixel. The value of each node in a high level is the weighted average of node values in the next lower level. Note that node spacing doubles from level to level, while the same weighting pattern or "generating kernel" is used to generate all levels.

The generating kernel: Note that the same 5-by-5 pattern of weights w is used to generate each pyramid array from its predecessor. This weighting pattern, called the generating kernel, is chosen subject to certain constraints. For simplicity we make w separable:

$$w(m,n) = \hat{w}(m)\hat{w}(n) \tag{4.18}$$

The one-dimensional, length 5, function $\hat{w}$ is normalized

$$\sum_{m=-2}^{2} \hat{w}(m) = 1 \tag{4.19}$$

and symmetric

$$\hat{w}(i) = \hat{w}(-i) \quad \text{for } i = 0, 1, 2. \tag{4.20}$$

An additional constraint is called equal contribution. This stipulates that all nodes at a given level must contribute the same total weight (=1/4) to nodes at the next higher level. Let $\hat{w}(0) = a, \hat{w}(-1) = \hat{w}(1) = b$, and $\hat{w}(-2) = \hat{w}(2) = c$ in this case equal contribution requires that $a + 2c = 2b$. These three constraints are satisfied when

$$\hat{w}(0) = a \tag{4.21}$$

$$\hat{w}(-1) = \hat{w}(1) = \frac{1}{4} \tag{4.22}$$

$$\hat{w}(-2) = \hat{w}(2) = \frac{1}{4} - \frac{a}{2} \tag{4.23}$$

Equivalent weighting functions: Iterative pyramid generation is equivalent to convolving the image g_0 with a set of "equivalent weighting functions" h_l :

$$g_l = h_l \oplus g_0 \tag{4.24}$$

or

$$g_l(i,j) = \sum_{m=-M_l}^{M_l} \sum_{n=-M_l}^{M_l} h_l(m,n) g_l\left((i2^l + m)(j2^l + n)\right) \tag{4.25}$$

where the size M_l of the equivalent weighting function doubles from one level to the next, as does the distance between samples.

Equivalent weighting functions for Gaussian pyramid levels 1, 2, and 3 are shown in Figure 4.17 a) and b). In this case a = 0.4. The shape of the equivalent function converges rapidly to a characteristic form with successively higher levels of the pyramid, so that only its scale changes. However, this shape does depend on the choice of a in the generating kernel. Characteristic shapes for four choices of a. Note that the equivalent weighting functions are particularly Gaussian-like when a = 0.4 When a = 0.5 the shape is triangular; when a = 0.3 it is flatter and broader than a Gaussian. With a = 0.6 the central positive mode is sharply peaked, and is flanked by small negative lobes. Figure 4.17 a) The equivalent weighting functions $\tilde{h}_l(x)$ for nodes in levels 1, 2, 3, and infinity of the Gaussian pyramid. Note that axis scales have been adjusted by factors of 2 to aid comparison here the parameter a of the generating kernel is 0.4, and the resulting equivalent weighting functions closely resemble the Gaussian probability density functions and Figure 4.17 b) The shape of the equivalent weighting function depends on the choice of parameter a. For a = 0.5, the function is triangular; for a = 0.4 it is Gaussian-like, and for a = 0.3 it is broader than Gaussian. For a = 0.6 the function is trimodal.

Example: The values of the weighting function is applied in the pyramid decomposition and reconstruction $w(m,n)$ is the 5-tap filter: $\left[\frac{1}{4} - \frac{a}{2}, \frac{1}{4}, a, \frac{1}{4}, \frac{1}{4} - \frac{a}{2}\right]$; $a = 0.4$ or

$[1 \quad 4 \quad 6 \quad 4 \quad 1]/16.$

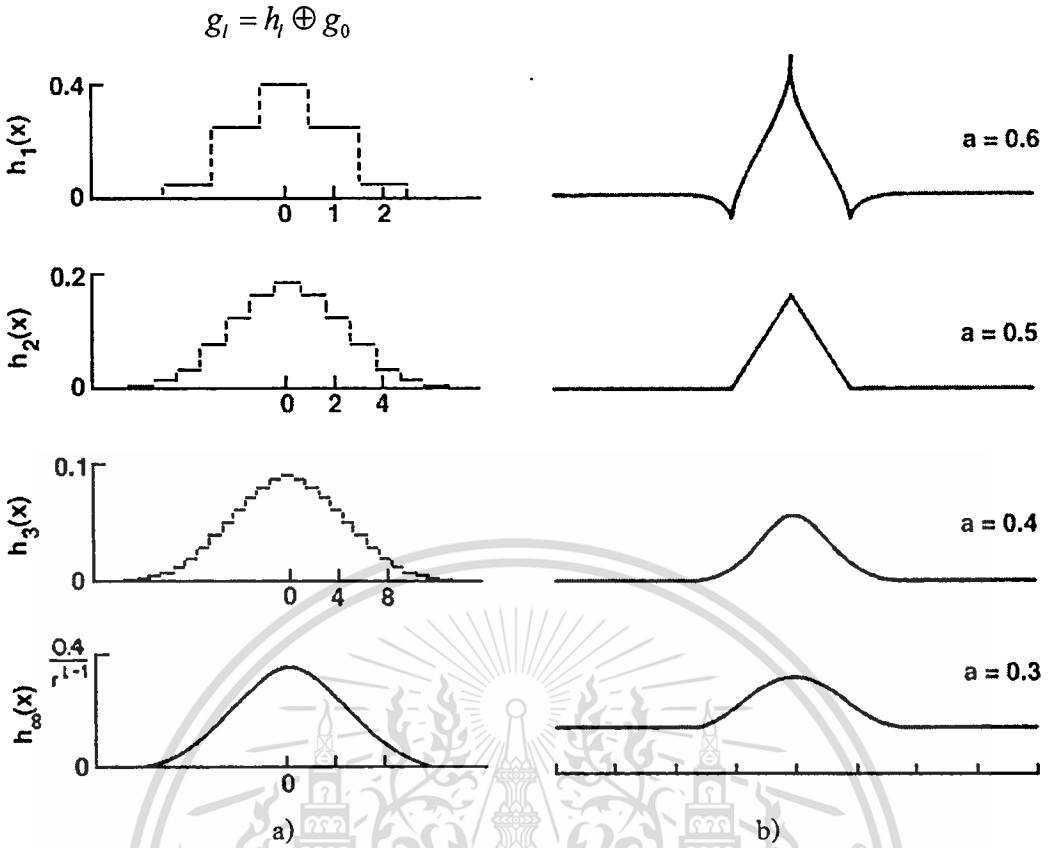


Figure 4.17 The shape of the equivalent weighting function.

2) Pyramid reconstruction

We now define a function *EXPAND* as the reverse of *REDUCE*. Thus, *EXPAND* applied to g_l of the Gaussian pyramid would yield an array $g_{l,1}$ which is the same size as g_{l-1}. Let $g_{l,n}$ be the result of expanding g_l, *n times*. Then $g_{l,0} = g_l$ and

$$g_{l,n} = EXPAND(g_l, n-1) \tag{4.26}$$

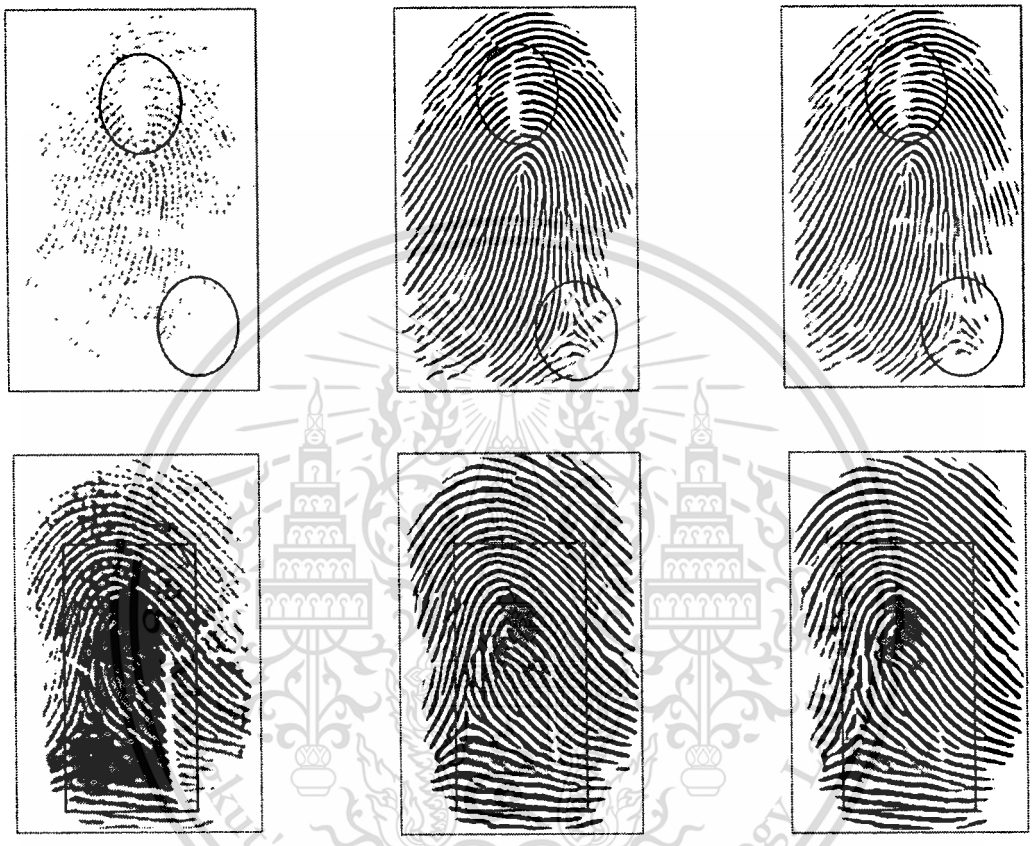
By EXPAND we mean, for levels $-2 < l < 2$ and $0 < n$

$$g_{l,n}(i,j) = 4 \sum_{m=-2}^{2} \sum_{n=-2}^{2} w(m,n)\, g_{l-1}\left(\frac{i-m}{2}, \frac{j-n}{2}\right) \tag{4.27}$$

Only terms for which $(i-m)/2$ and $(j-n)/2$ are integers are included in this sum.

If we apply EXPAND *l times* to image g_l, we obtain $g_{l,1}$, which is the same size as the original image g_0. After decomposed (reduced) and applied directional filtering (Gabor filter / Second derivative of Gaussian filter) directly to the image for ridge directional, at each level the

image L_1, L_2, L_3 are obtained. We reconstructed in Figure 4.15 shown images $g_{3,2}, g_{2,1}$, and $g_{1,0}$ obtained by expanding levels of the Gaussian pyramid. (See Figure 4.18) the directional filtering effect of the pyramid technique is now shown clearly that includes: First column a) original images, second column b) corresponding to Gabor filter (GBF) and third column c) corresponding to second derivative of Gaussian filter (SDGF).



a) Original images b) Pyramid with GBF c) Pyramid with SDGF

Figure 4.18 Example of fingerprint enhancements is applied pyramid technique.

4.3.4 Fingerprint enhancement using directional wavelet transform

The wavelet transform is a transform similar to the Fourier transform (or much more to the windowed Fourier one as we see later) with a completely different merit function. The main difference is this: Fourier transform decomposes the signal into sine and cosines i.e. the functions localized in Fourier space; in contrary the wavelet transform uses functions that are localized in both the real and Fourier space.

1) Wavelet decomposition

Wavelet transform is suited for the analysis of transient and time varying signals. In two dimensions, a scaling function $\varphi(x,y)$, and three directional wavelets $\psi^H(x,y), \psi^V(x,y)$ and $\psi^D(x,y)$ are necessary. Each scaling function or wavelet is the product of the one dimensional scaling function φ and corresponding wavelet ψ. The four two-dimensional products produce the scaling function (4.28) and separable directional sensitive wavelets (4.29), (4.30) and (4.31).

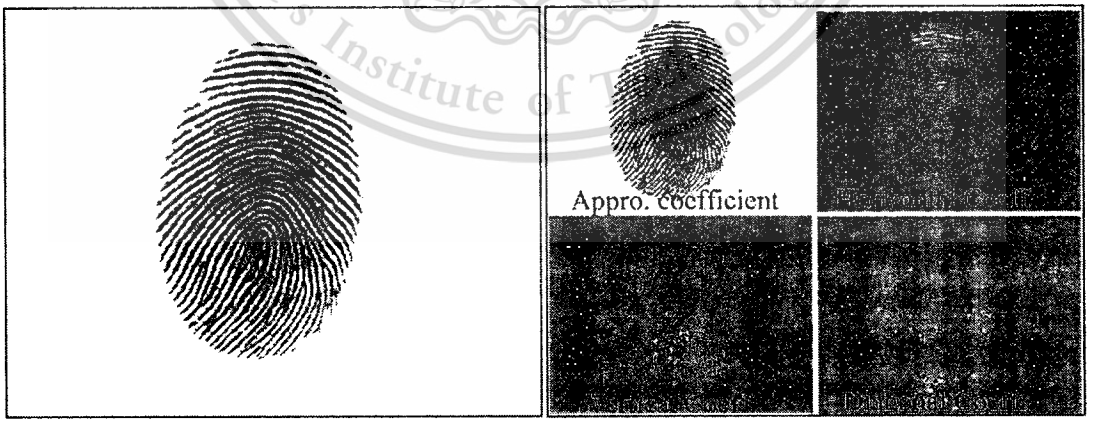
$$\varphi(x,y) = \varphi(x)\varphi(y) \tag{4.28}$$

$$\psi^H(x,y) = \psi(x)\varphi(y) \tag{4.29}$$

$$\psi^V(x,y) = \varphi(x)\psi(y) \tag{4.30}$$

$$\psi^D(x,y) = \psi(x)\psi(y) \tag{4.31}$$

These wavelets measure the gray level variations for images along three directions, where $\psi^H(x,y)$ measures variations along columns (horizontal), $\psi^V(x,y)$ responds to variations along rows (vertical) and $\psi^D(x,y)$ corresponds to variations along diagonals. The decomposition structure of an image is shown in Figure 4.19 b), where the upper left quadrant is the approximate of the original image, while the upper right, lower left and right quadrants represent the vertical, horizontal and diagonal details of the original at its nearest decomposition level.



a) Original image b) Decomposition one level

Figure 4.19 Wavelet transform one level.

The existing directional sensitivity does not change computational complexity of the two dimensional wavelet transform. In a two dimensional discrete wavelet transform, the scaled and translated basis functions are defined by:

$$\varphi_{j,m,n}(x,y) = 2^{j/2}\varphi(2^j x - m, 2^j y - n), \tag{4.32}$$

$$\psi^i_{j,m,n}(x,y) = 2^{j/2}\psi^i(2^j x - m, 2^j y - n), \quad i = \{H,V,D\}, \tag{4.33}$$

where index i identifies the directional wavelets according to equation (4.29), (4.30) and (4.31). The discrete wavelet transform of function $f(x,y)$ of size $M \times N$ is formulated as:

$$W_\varphi(j_0,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\varphi_{j_0,m,n}(x,y) \tag{4.34}$$

$$W^i_\psi(j,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\psi^i_{j,m,n}(x,y) \quad ;i = \{H,V,D\} \tag{4.35}$$

where $i = \{H,V,D\}$, j_0 is the starting scale, the $W_\varphi(j_0,m,n)$ coefficients define the approximation of $f(x,y)$, at scale j_0. The $W_\psi(j,m,n)$ coefficients represent the horizontal, vertical and diagonal details for scales $j \geq j_0$. Here $j_0 = 0$ and select $M + N = 2^J$ so that $j = 0,1,2,..,J-1$ and $m,\ n = 0,1,2,..,(2^j - 1)$. Then $f(x,y)$ is obtained via the inverse discrete wavelet transform.

$$f(x,y) = \frac{1}{\sqrt{MN}}\sum_m\sum_n W_\varphi(j_0,m,n)\varphi_{j_0,m,n}(x,y) + \frac{1}{\sqrt{MN}}\sum_{i=H,V,D}\sum_{j=j_0}^{\infty}\sum_m\sum_n W^i_\psi(j,m,n)\psi^i_{j,m,n}(x,y), \tag{4.36}$$

Now the wavelets are defined by both the scaling function $\varphi(x,y)$ (Father Wavelet) and wavelet functions $\psi(x,y)$ (mother wavelet) in the discrete time domain. The wavelet function acts as a bandpass filter whose bandwidth is reduced to half after each scaling. At the level 1 of the wavelet transform, the original image is divided into four parts: approximation, horizontal detail, vertical detail and diagonal detail (see Finguer4.19). The scaling function filters the lowest level of the transform to ensure the 2D series formulation of the inverse discrete wavelet transform. The 2D wavelet transform has been implemented as a multiple level transformation. At the level one, rows of $f(x,y)$ are lowpass and highpass filtered and downsampled. Then columns of the row filtered images are lowpass and highpass filtered and downsampled similarly. The two dimensional discrete wavelet transform filters the scaling approximation coefficients to construct the scale approximation and detail coefficients (discussed

in Chapter 3). The output of each level always includes: approximation, horizontal detail, vertical detail and diagonal detail. As a result, each of them is a quarter of the size of its original image. Four quarter size output sub-images $(W_\varphi, W_\psi^H, W_\psi^V, W_\psi^D)$ are the inner products of $f(x,y)$ and the 2D scaling and wavelet functions, followed by downsampling by a factor of two.

In this thesis, we select Daubechies 4 (details of db4 discussed in Chapter 3) to implement the decomposition for its sufficient information in sub-image approximation. Theoretically, we can decompose the image into sub-images at any level. However, too low resolution is not suitable because an excessive down sampling of the signal can vanish the orientation characteristic of the ridge structure. Generally, several decomposition levels are selected. We used only one decomposition level in this presents. The original fingerprint images is decomposed into approximation and detail sub-images, at each sub-dimension is applied directional filtering (Gabor filter/Second derivative of Gaussian filter) to tune up the image features and obtained directional filtering (DF) decomposed images are shown in Figure 4.19, the sub-image shows: horizontal, vertical and diagonal component respectively.

The wavelet transforms have proven to be an effective tool in the denoising. This possible due to fact that when wavelet transform is applied to an image it is transformed into a compact set of coefficients which can further be used for easier computation. Wavelet bases are bases of nested function spaces, which can be used to analyze signals at multiple scales. Wavelet coefficients contain both time and frequency information as the basis functions varies in position and scale. The coefficients can be obtained from the image by applying the Discrete Wavelet Transform where in the output of the low pass filter is the area of interest (discussed Chapter 3).

2) Wavelet Reconstruction

After modifying the approximation sub-image with directional filtering (Gabor filters / the second derivative of Gaussian filter), the final step, an image of the enhancement is reconstructed. Normally, the wavelets used in reconstruction are the same as that used in the decomposition process. Figure 4.20 shown fingerprint enhancements applied directional wavelet transform includes that Gabor filter and Second derivative of Gaussian filter includes that: first column a) original images, second column b) corresponding to Gabor filter (GBF) and third column c) corresponding to the Second derivative of Gaussian filter (SDGF).

a) Original images b) Wavelet with GBF c) Wavelet with SDGF

Figure 4.20 Example of fingerprint enhancements is applied directional wavelet transform.

4.4 Minutiae Extraction

After a fingerprint image has been enhanced, the next step is to extract the minutiae from the enhanced image. Most automatic systems for fingerprint comparison are based on minutiae matching; hence, reliable minutiae extraction is an extremely important task and a substantial amount of research has been devoted to this topic. Most of the proposed methods require the fingerprint gray-scale image to be converted into a binary image. Some binarization processes greatly benefit from an a priori enhancement; on the other hand, some enhancement algorithms directly produce a binary output, and therefore the distinction between enhancement and binarization is sometimes faded. The binary images are usually submitted to a thinning stage which allows for the ridge line thickness to be reduced to one pixel, resulting in a skeleton image (Figure 4.21). A simple image scan then allows the detection of pixels that correspond to minutiae [1].

4.4.1 Binarization-based methods

The general problem of image binarization has been widely studied in the fields of image processing and pattern recognition. The simplest approach uses a global threshold t and works by setting the pixels whose gray-level is lower than t to 0 and the remaining pixels to 1. In general, different portions of an image may be characterized by different contrast and intensity and, consequently, a single threshold for the entire image is not sufficient for a correct binarization. For this reason, the local threshold technique changes t locally, by adapting its value to the average local intensity. In the specific case of fingerprint images, which are sometimes of very poor quality, a local threshold method cannot always guarantee acceptable results and more effective fingerprint-specific solutions are necessary. In the rest of this section, the commonly used binarization methods used for fingerprints are briefly summarized.

a) Original image b) binary of enhancement c) Thinning

Figure 4.21 Results of applying binarization and thinning directly to the original image with enhancement.

The most commonly employed method of minutiae extraction is the Crossing Number (CN) concept [86]. This method involves the use of the skeleton image where the ridge flow pattern is eight-connected. The minutiae are extracted by scanning the local neighbourhood of each ridge pixel in the image using a 3x3 window. The CN value is then computed, which is defined as half the sum of the differences between pairs of adjacent pixels in the eight neighbourhood. Using the properties of the CN as shown in Table 4.1 and Figure 4.22, the ridge pixel can then be classified as a ridge ending, bifurcation or non-minutiae point. For example, a ridge pixel with a CN of one corresponds to a ridge ending, and a CN of three corresponds to a bifurcation.

Table 4.1 Properties of the Crossing Number.

CN	Property
0	Isolated point
1	Ridge ending point
2	Continuing ridge point
3	Bifurcation point
4	Crossing point

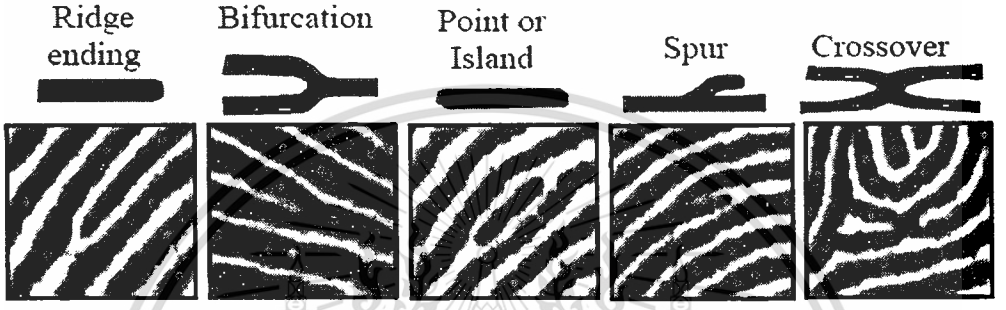


Figure 4.22 Five most common minutiae types (black line)

Other authors [78] have also performed minutiae extraction using the skeleton image. Their approach involves using a 3x3 window to examine the local neighborhood of each ridge pixel in the image. A pixel is then classified as a ridge ending if it has only one neighboring ridge pixel in the window, and classified as a bifurcation if it has three neighboring ridge pixels. Consequently, it can be seen that this approach is very similar to the Crossing Number method.

4.4.2 Minutiae detection

The Crossing Number (CN) method is used to perform minutiae extraction. This method extracts the ridge endings and bifurcations from the skeleton image by examining the local neighborhood of each ridge pixel using a 3x3 window. The CN for a ridge pixel P is given by:

$$CN = 0.5\sum_{i=1}^{8}\left|P_i - P_{i+1}\right|, \qquad P_9 = P_1 \tag{4.37}$$

where P_i is the pixel value in the neighborhood of P. For a pixel P, its eight neighboring pixels are scanned in an anti-clockwise direction as follows Table 4.2:

Table 4.2 Crossing Number computes in the eight neighboring pixels of minutiae detection.

P_4	P_3	P_2
P_5	P	P_1
P_6	P_7	P_8

After the CN for a ridge pixel has been computed, the pixel can then be classified according to the property of its CN value. As shown in Figure 4.23, a ridge pixel with a CN of one corresponds to a ridge ending, and a CN of three corresponds to a bifurcation and Figure 4.24 a) a Crossing Number of one corresponds to a ridge ending pixel. b) a Crossing Number of three corresponds to bifurcation pixel of minutiae detection.

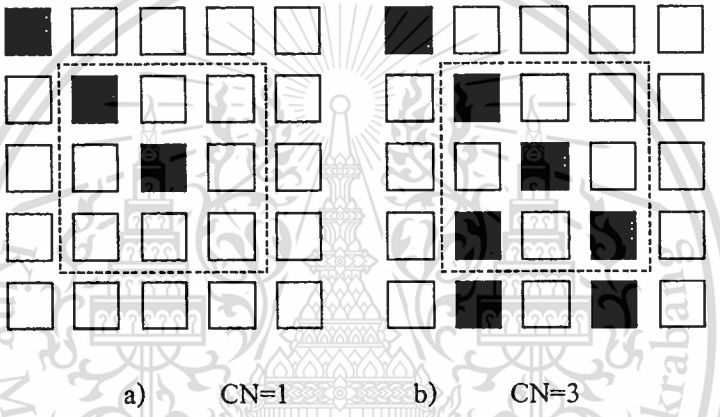


Figure 4.23 Examples of a ridge ending and bifurcation pixel.

Figure 4.24 Minutiae detection, circles red and green corresponding to ridges ending and ridges bifurcation.

4.5 Evaluation of Enhancement Performance

4.5.1 Core point detection techniques

Most of the approaches proposed in the literature for singularity detection operate on the fingerprint orientation image. In the rest of this section, the main approaches are coarsely classified and a subsection is dedicated to each family of algorithms. The core point is generally recognized as the top-most point that the ridges making turns. In order to detect the fingerprint center point area, we first locate the core point corresponding to the uppermost point contained in the inner-most ridge line. There exist several techniques for core point detection. They are; for instance; Poincare index (PC), Direction of curvature (DC), and Geometry region (GR). Each holds its individual complexity and performance [1].

4.5.2 Poincare index technique (PC)

The Poincare one is fairly simple and suitable for both core point and delta point identify both core point and delta point. Upon the availability of estimated orientation field $\theta(i,j)$ given above, for the pixel in the sub block centered at (i,j) we can compute Poincare index [79, 80, 81]. Let **G** be a vector field and C be a curve immersed in **G**; then the Poincaré index $P_{G,C}$ is defined as the total rotation of the vectors of **G** along C (see Figure 4.25).

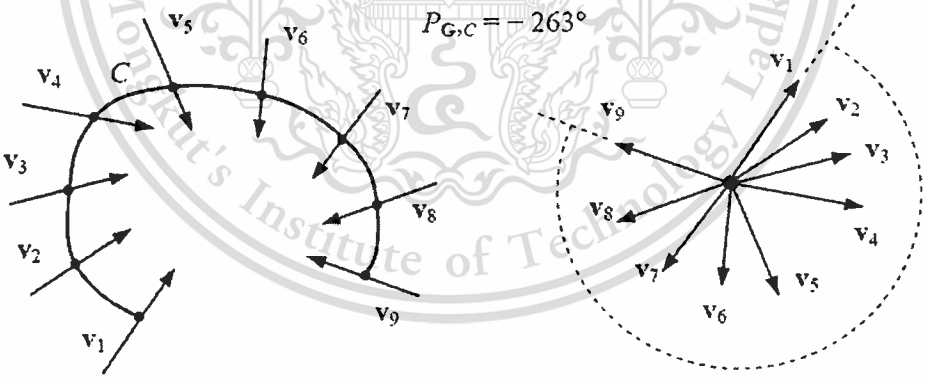


Figure 4.25 The Poincaré index computed over a curve C immersed in a vector field G.

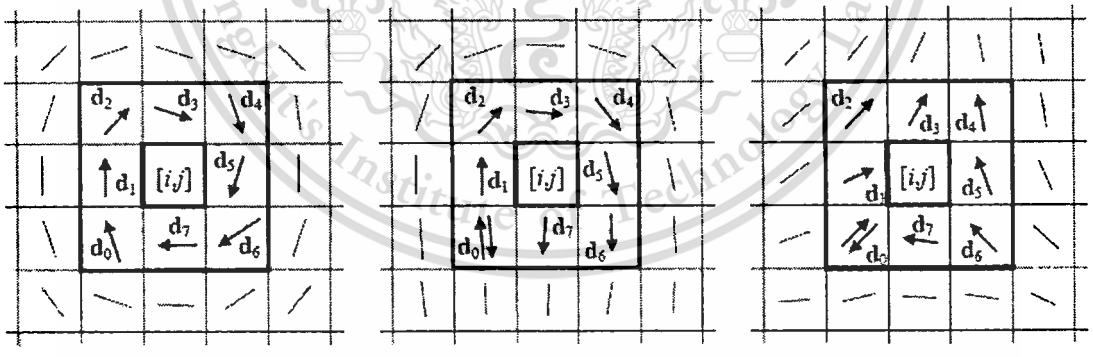
Let **G** be the discrete vector field associated with a fingerprint orientation image θ and let $[i,j]$ be the position of the element θ(i, j) in the orientation image; then the Poincaré index $P_{G,C}(i,j)$ at $[i,j]$ is computed as follows. The curve C is a closed path defined as an ordered sequence of some elements of θ, such that $[i,j]$ is an internal point.

$P_{G.C}(i,j)$ is computed by algebraically summing the orientation differences between the adjacent elements of C. Summing orientation differences requires a direction (among the two possible) to be associated at each orientation. A solution to this problem is to randomly select the direction of the first element and assign the direction closest to that of the previous element to each successive element. It is well known and can be easily shown that, on closed curves, the Poincaré index assumes only one of the discrete values: 0°, ± 180°, and ± 360°. In the case of fingerprint singularities:

$$P_{GC}(i,j)=\begin{cases} 0° & \text{if } [i,j] \text{ does not belong to any singular region} \\ 360° & \text{if } [i,j] \text{ belong to a whorl type singular region} \\ 180° & \text{if } [i,j] \text{ belong to a loop type singular region} \\ -180° & \text{if } [i,j] \text{ belong to a delta type singular region} \end{cases} \tag{4.38}$$

Figure 4.26 shows three portions of orientation image. The path defining C is the ordered sequence of the eight elements $\mathbf{d}$ (k = 0...7) surrounding $[i,j]$. The direction of the elements $\mathbf{d}$ is chosen as follows: $\mathbf{d}$ is directed upward; $\mathbf{d}$ (k = 1...7) is directed so that the absolute value of the angle between $\mathbf{d}_k$ and $\mathbf{d}_{k-1}$ is less than or equal to 90°. The Poincaré index is then computed as

$$P_{GC}(i,j)=\sum_{k=0}^{7} angle(d_k, d_{(k+1) \bmod 8}) \tag{4.39}$$



$P_{G.C}(i,j) = 360°$ $P_{G.C}(i,j) = 180°$ $P_{G.C}(i,j) = -180°$

Figure 4.26 Examples of Poincaré index computation in the 8-neighborhood of points belonging (from left to right) to a whorl, loop, and delta singularity, respectively.

Note that, for the loop and delta examples (center and right), the direction of d_0 is first chosen upward (to compute the angle between d_0 and d_1) and then successively downward (when computing the angle between d_7 and d_0)

a) Core point detected without enhancement b) Core point detected with enhancement

Figure 4.27 Examples of Poincaré index technique detection (star) on the fingerprint image.

4.6 Summary

The performance of a fingerprint feature extraction and matching algorithms depend heavily upon the quality of the input fingerprint image. In this chapter we presented fingerprint image enhancement algorithm with the focus of directional filtering. The insertion of the basic directional filters into the more sophisticate algorithms has several advantages over the single application of individual scheme. The enhancement utilizes the full contextual information (Gabor filter, Second derivative of Gaussian filter, pyramid technique and directional wavelet transform). Regardless of the complexity of the algorithm, these enhancement methods offer considerable image improvement. The experiments detailed in the next chapter reveal the said arguments.

Chapter 5

Experiment & Results

5.1 Preprocessing and Post Process

All the methods and algorithms described in this dissertation were implemented using MATLAB V7.1 on the Laptop, Microsoft Windows Vista™ Home Premium. The experiments were performed on a Centrino 2.16 GHz with 3GB of RAM. When testing the performance of the enhancement algorithm, the computational time was differenced measured for each enhancement techniques. The experimental results section is to show the results of each stage in the enhancement algorithm. We used the downloaded a fingerprint database from the DB1_A of FVC2004 (Fingerprint Verification Competition 2004) is used to test the experiment performance. There are 800 fingerprint images captured with optical sensor, "V300" by CrossMatch. An image size is of 640×480 pixels with 500 dpi resolutions. With eyes observation, we found that 746 images hold core point. The performance of the filter is measured by the success in core point locating for core point detection.

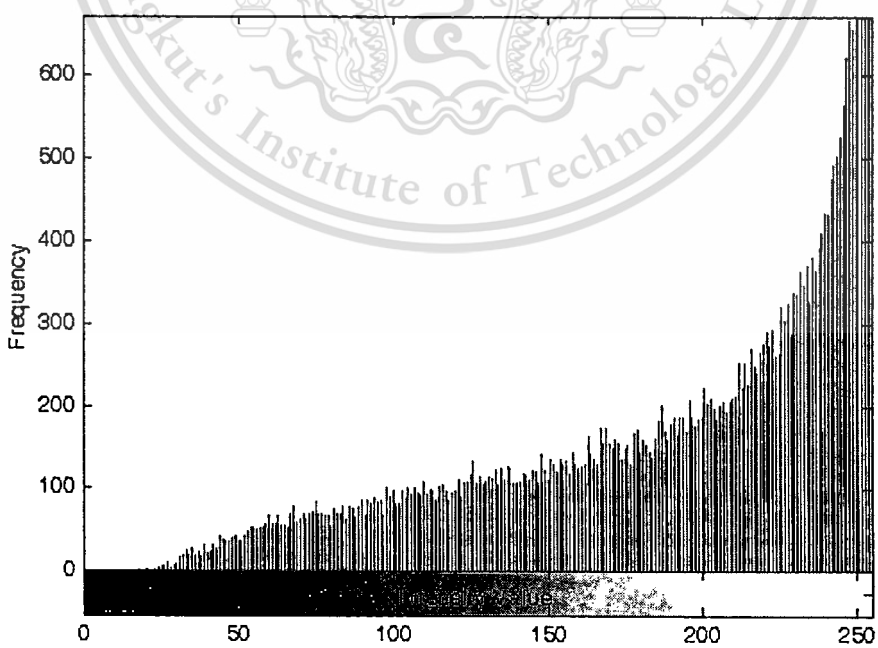
First stage the experimental results of fingerprint preprocessing includes: Normalization, Segmentation, Orientation field smoothing, ridge frequency estimation. Second stage the experimental results of fingerprint enhancement with directional filtering that mainly our method includes: Gabor filter, Second derivative of Gaussian filter, Pyramid technique and Directional wavelet transform. Last stage performance an algorithm used for core point detection, a Poincare technique is used.

5.1.1 Normalization

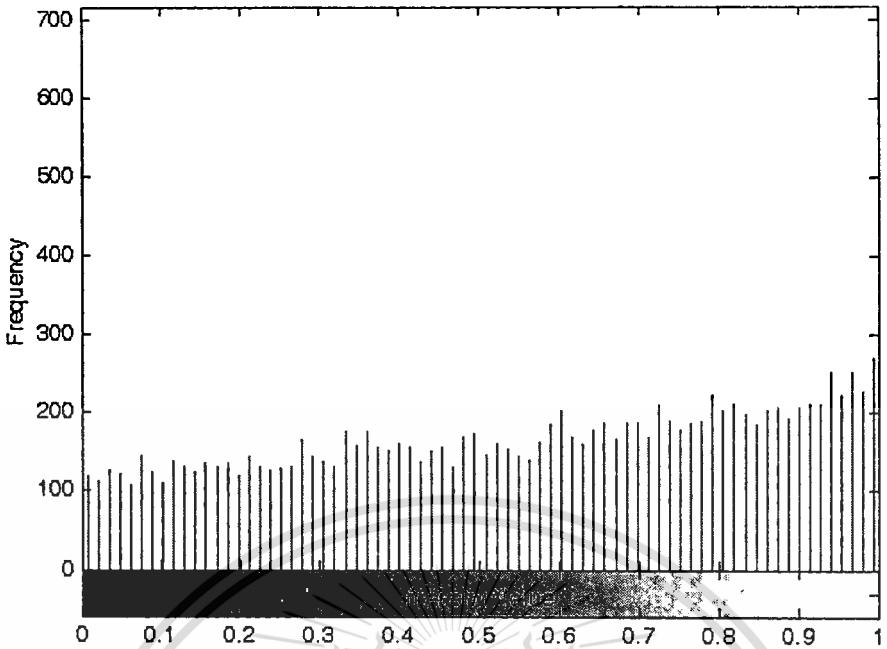
For all images submitted to the image enhancement process: a desired mean value of 0.5 and a variance of 1 are used for normalization. Therefore, each image is normalized to a predetermined level before proceeding on to the subsequent enhancement stages. Figure 5.1 shows the results of normalizing a fingerprint image so that it has a desired mean of 0.5 and a variance of 1. The histogram of the original image (see in Figure 5.1 c)) illustrates that increasing

the intensity values lie from on the left hand side to on the right hand side of the 0 – 255 scale. This results in the image having a high brightness, as shown in Figure 5.1 a). On the other hand, the histogram of the normalized image (see in Figure 5.1 d)) shows that the range of intensity values has been adjusted such that there is a more balanced distribution between the dark and light pixels. Hence, normalizing the image improves the contrast between the ridges and valleys, as shown in Figure 5.1 b). Additionally, the histograms plots in Figure 5.1 c) and Figure 5.1 d) show that the normalization process does not alter the shape of the original histogram plot; only the relative position of the values along the *x* axis is shifted, which means the structure of the ridges and valleys are not changed.

a) Original image b) Normalized image



c) Histogram of the original image



d) Histogram of the normalized image

Figure 5.1 The result of normalization using a desired mean and variance of 0.5 and 1.

5.1.2 Segmentation

As shown in Figure 5.2 the results of segmenting a fingerprint image based on mean thresholding of image. Hence, a mean threshold of image is used to separate the fingerprint foreground area from the background regions. The final segmented image is formed by assigning the regions with a variance value below the threshold to a grey-level value of mean of image, as shown in Figure 5.2 b).These results show that the foreground regions segmented by this method comprise only of areas containing the fingerprint ridge structures, and that regions are not incorrectly segmented. Hence, the mean thresholding method is effective in discriminating the foreground area from the background regions.

There is a trade-off involved when determining the threshold value used to segment the image. If the threshold value is too large, results have shown that foreground regions may be incorrectly assigned as background regions. Conversely, if the threshold value is too small, background regions may be mistakenly assigned as part of the fingerprint foreground area. Hence, a mean threshold of image gives the optimal results in terms of differentiating between the foreground and background regions.

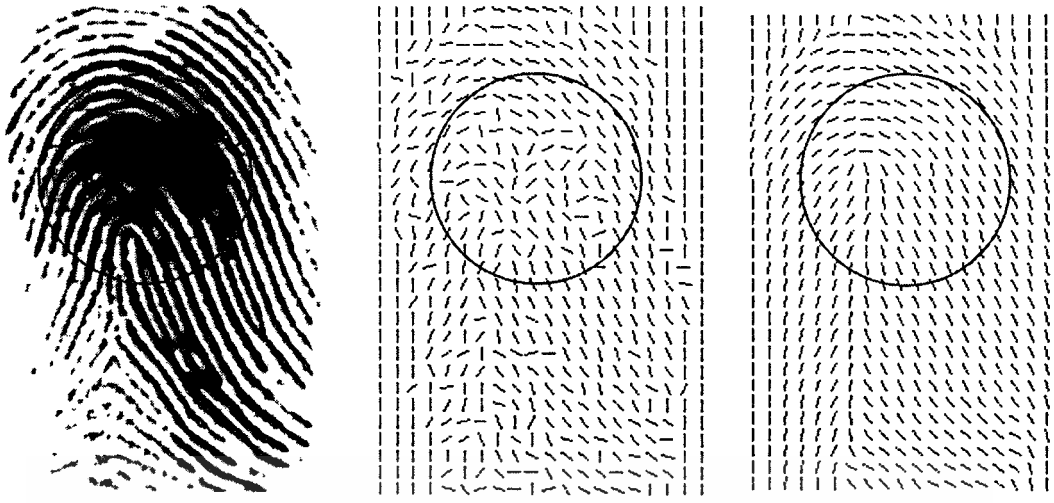
a) Original image b) Segmentation

Figure 5.2 The result of segmentation using a mean threshold of image and a block size of 15×15

5.1.3 Orientation field estimation

The directional filtering (Gabor filtering and Second derivative of Gaussian filter) stages of the enhancement process relies heavily on filtering along the local ridge orientation (field estimation) in order to enhance the ridge structure and reduce noise. Hence, it is important to obtain an accurate estimation of the orientation field. As the orientation estimation stage plays a central role in the enhancement process, I have conducted an extensive series of experiments to evaluate the performance of the orientation field estimation algorithm in Figure 5.3 b).

The discontinuity of ridge and valley due to noise can be softening by applying a low pass filter (2-D Gaussian filter), on the other hands, depends very much on the orientation and ridge frequency. Because the local orientation changes very rapidly in the core point area, we almost cannot make it accurate. So, the result of enhancement in the core point is fairly poor. There comes the need of prior field smoothing. The window size used the field estimation process is also important in such a way that; on one side too big window can give the good field smoothing. Filtering effect becomes stable when the window size is 31× 31 pixels as shown in Figure 5.3 c).

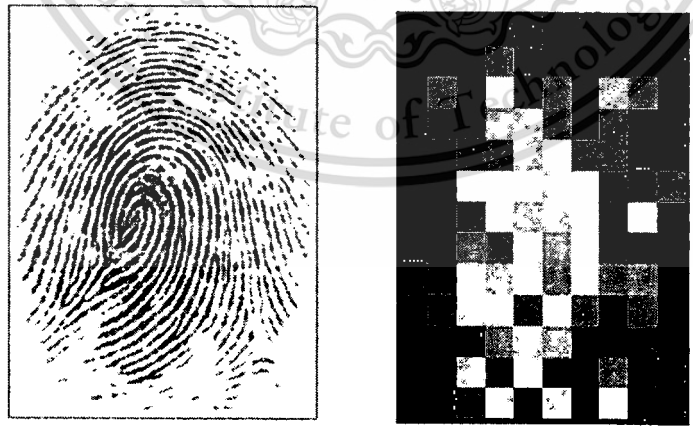


a) Original image b) Field estimation c) Field smoothing

Figure 5.3 Orientation field smoothing.

5.1.4 Ridge frequency estimation

Together with the ridge orientation, the ridge frequency is another important parameter used in the construction of the directional filtering (Gabor filter/second derivative of Gaussian filter). Note that the results for the ridge frequency values will be presented in terms of ridge wavelength for easier interpretation of the results. For example, if the ridge spatial frequency value is 1/10 pixels, then the results will present this as a ridge wavelength of 10. Fingerprint images as Figure 5.4 illustrates the ridge wavelength image.



a) Original image b) Ridge wavelength

Figure 5.4 The estimated ridge wavelength (ridge frequency).

5.2 Directional Filtering

The configurations of parallel ridges and valleys with well-defined frequency and orientation in a fingerprint image provide useful information which helps in removing undesired noise. The sinusoidal-shaped waves of ridges and valleys vary slowly in a local constant orientation. Therefore, a bandpass filter that is tuned to the corresponding frequency and orientation can efficiently remove the undesired noise and preserve the true ridge and valley structures. As shown in Figure 5.5 data flow diagram fingerprint enhancement involve: Normalization, segmentation, Field estimation, Field smoothing, Ridge frequency estimation and Gabor filtering.

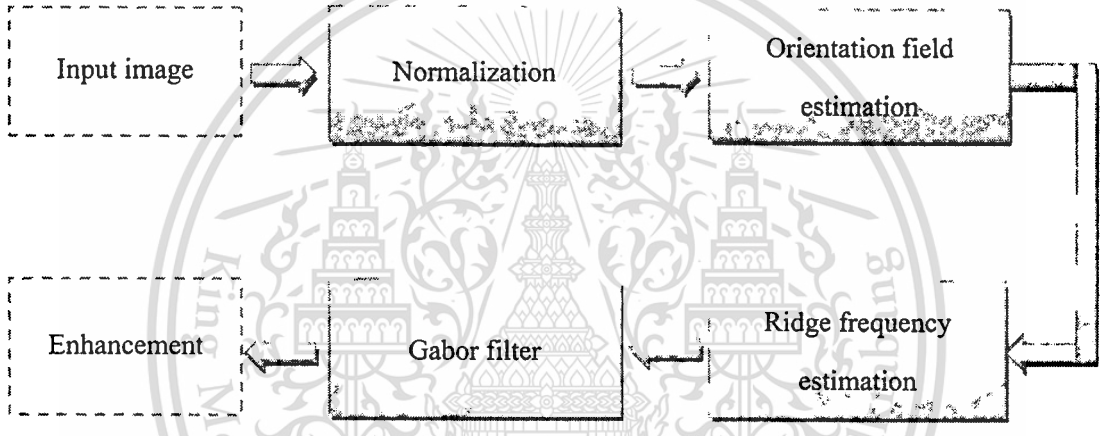


Figure 5.5 Data flow diagram fingerprint enhancement using Gabor filtering.

5.2.1 Gabor filtering

This is the stage that performs the actual enhancement of the fingerprint image. The propose of the directional filtering stage is to enhance the clarity of the ridge structures while reducing noise in the image, in order to assess the performance of the Gabor filtering. The Gabor filters parameters σ_x and σ_y controls the bandwidth of the filter, and must be chosen carefully as they have a significant effect on the enhancement results. The value of σ_x determines the degree of contrast enhancement between ridges and valleys, and σ_y determines the amount of smoothing applied to the ridges along the local orientation. Figure 5.6 illustrates the results of using the same values of σ_x and σ_y to apply the Gabor filter to a fingerprint image.

a) Original image b) $\sigma_x = \sigma_y = 5\left(k_x = k_y = 0.5\right)$

Figure 5.6 Enhancement results by using the Gabor filter.

5.2.2 Second derivative of Gaussian filtering

As shown in Figure 5.7, the steps of experiment second derivative of the Gaussian filter as band-pass filter that is tuned to the corresponding frequency and orientation can efficiently remove the undesired noise while preserving the true ridge and valley structures (see Figure 5.8).

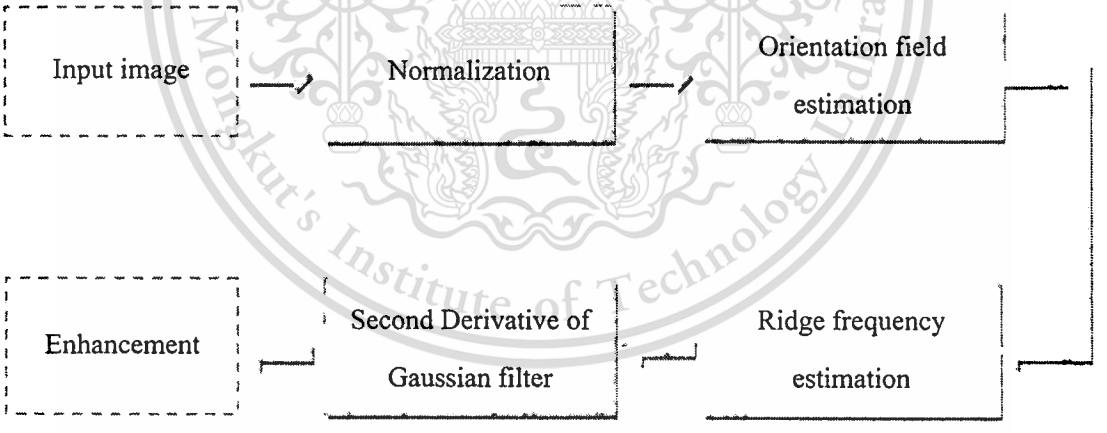


Figure 5.7 Data flow diagram fingerprint enhancements using the second derivative of Gaussian filter.

Second derivative of Gaussian filter is similar Gabor filters to remove the artifacts and noise and preserve true ridge and valley structures. To apply second derivative of Gaussian filter to an image, the parameters must be specified σ_x and σ_y controls the bandwidth of the filter are similar Gabor filters (see in section 5.2.1), and must be chosen carefully as they have a significant

effect on the enhancement results. On the other hand, the smaller the values of σ_x and σ_y, the less likely the filters will be faded ridges and valleys; consequently, they will be less effective in removing the noise as shown in Figure 5.8 the optimal values of the second derivative of Gaussian filter for different parameter values of σ_x and σ_y have shown that using $k_x = 0.4$ and $k_y = 0.5$.

a) Original image b) $\sigma_x = 4, \sigma_y = 5 \left(k_x = 0.4, k_y = 0.5\right)$

Figure 5.8 Enhancement results by using the second derivative of Gaussian filter.

5.2.3 Pyramid Technique

As illustrate Figure 5.9 steps of experiment pyramid technique the original image g_0 to obtain an image g_1. We say that g_1 is a "reduced" version (divide by factor two) of g_0. In a similar way we form g_2 as a reduced version of g_1, and last level form g_3 as a reduced version of g_2. At each level, the directional filter (Gabor filter/Second derivative of Gaussian filter) is applied directly to the image for ridge directional and the image L_1, L_2, L_3 are obtained. So that, we reconstructed images $g_{3,2}, g_{2,1}$, and $g_{1,0}$ obtained by expanding levels of the Gaussian pyramid. The directional filtering effect of the pyramid technique is now shown clearly (see Figure 5.10).

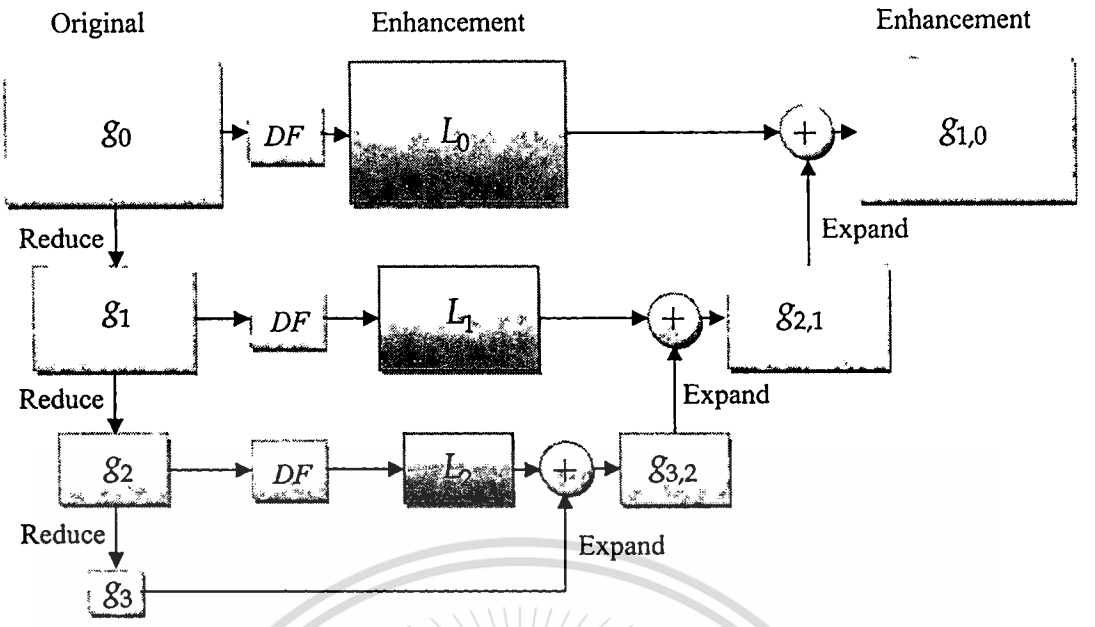


Figure 5.9 Data flow diagram of experiment of fingerprint enhancement by using pyramid technique.

a) Original image b) Pyramid with ridge enhancement

Figure 5.10 Example of fingerprint enhancements result by using pyramid technique.

5.2.4 Directional wavelet transform

In our method, the Daubechies wavelet (db4) is applied, the directional wavelet transform is presented to investigate some critical aspects of fingerprint identification. At the level one of the wavelet transform, the original image is divided into four parts: approximation and sub-images that: horizontal, vertical and diagonal, where the size of each part is reduced by a downsampling factor of two before, directional filtering (the Gabor filter/second derivative of Gaussian filter) is applied directly to each sub-image. The wavelet transform and coefficient have

been employed in image processing and reconstruction the fingerprint image by using the enhanced approximation image and detail images produced in decomposition. The detail process of the enhancement is as shown Figure 5.11 and Figure 5.12 the directional wavelet transform and reconstruction with in fingerprint image enhancement

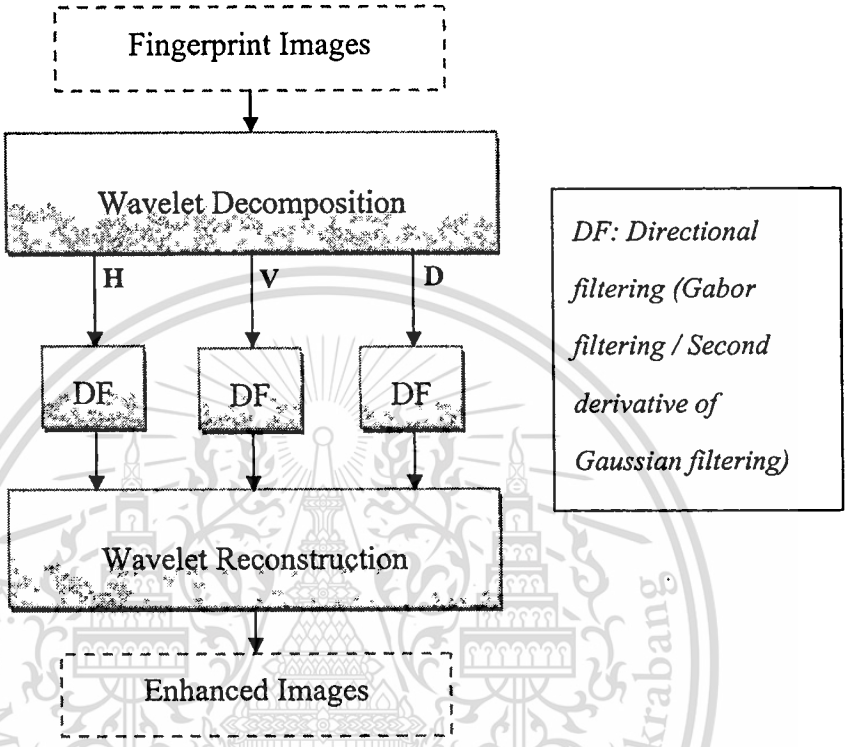
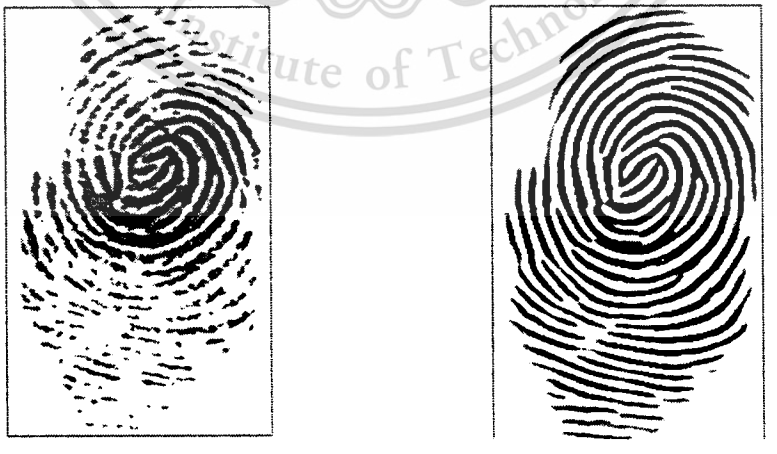


Figure 5.11 Data flow diagram of experiment fingerprint enhancement using directional wavelet transform.



a) Original image b) Wavelet reconstruction with ridge enhancement

Figure 5.12 Shown example of directional wavelet transform and reconstruction with in fingerprint image enhancement.

5.3 Post Process

The performance of fingerprint image enhancement with directional filtering technique is evaluates by two means; the success in core point detection and the consistency of minutiae. These two issues are discussed now.

5.3.1. Minutiae extraction

After a fingerprint image has been enhanced, the next step is to extract the minutiae from the enhanced image. The Crossing Number (CN) technique is able to extract the minutiae from the skeleton image. The minutiae validation algorithm is then evaluated to see how effective the algorithm is in detecting the true or false minutiae. The filtered image is thinned for the ease of ridge end and bifurcation observation. Figure 5.13 illustrates the results of extracting minutiae from fingerprint image. Hence, it can be shown that the CN technique is able to accurately detect all valid bifurcations and ridge endings from the skeleton image.

Figure 5.13 Shown the results of extraction minutiae fingerprint image.

5.3.2 Poincaré index techniques

The performance of the filter is also measured by the success in core point (singular point) detection using Poincare technique. A result of singularities detected by the Poincare technique method is shown in Figure 5.14 a) Good quality fingerprint and b) poor quality fingerprint (stars highlight the false singularities) or low-quality fingerprints is difficult and the Poincaré method may lead to the detection of false singularities. However, this method is simple and widely used.

a) b)

Figure 5.14 Core point detection by using Poincaré index technique for locating core points.

5.4 Experiments

With eyes observation, we found that 746 images actually hold core points. Without any application of enhancement process, we can feed these 746 images into the Poincare procedure for core point detection. There is no need to perform directional filtering to this set of the images. In contrast, for those that require further enhancement, for each enhancement techniques we did apply ridge frequency estimation, directional filtering (Gabor filter/Second derivative of Gaussian filter) are also applied in the scheme of pyramid technique and directional wavelet transform before, reducing the images is basically low pass filtering to prevent aliasing (noise) core point detection.

5.4.1 Experiment setup

As a matter of fact, those 746 images, 185 images hold 2 core points (dual core) and 561 images hold single core point (many of them in this group hold left-loop and right-loop patterns). With this observation, we performed the experiment separately. No enhancement and enhancing with conventional Gabor filtering are considered as conventional and basic result of core point identification. On another hand, application of several type of filters detailed previously are the main work of this thesis that we want to compare our proposal and achievements to the basic obtaining. The set up are shown in Figure. 5.15 and 5.16, respectively.

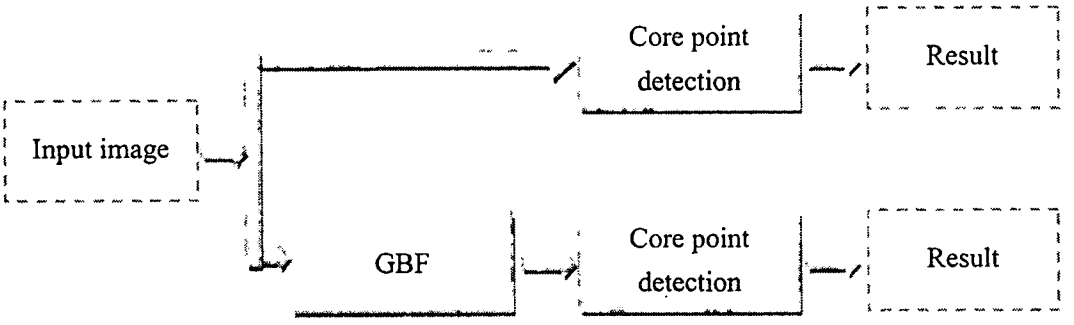


Figure 5.15 Data flow diagram of fingerprint core point detection with none enhancement and conventional Gabor filtering enhancement.

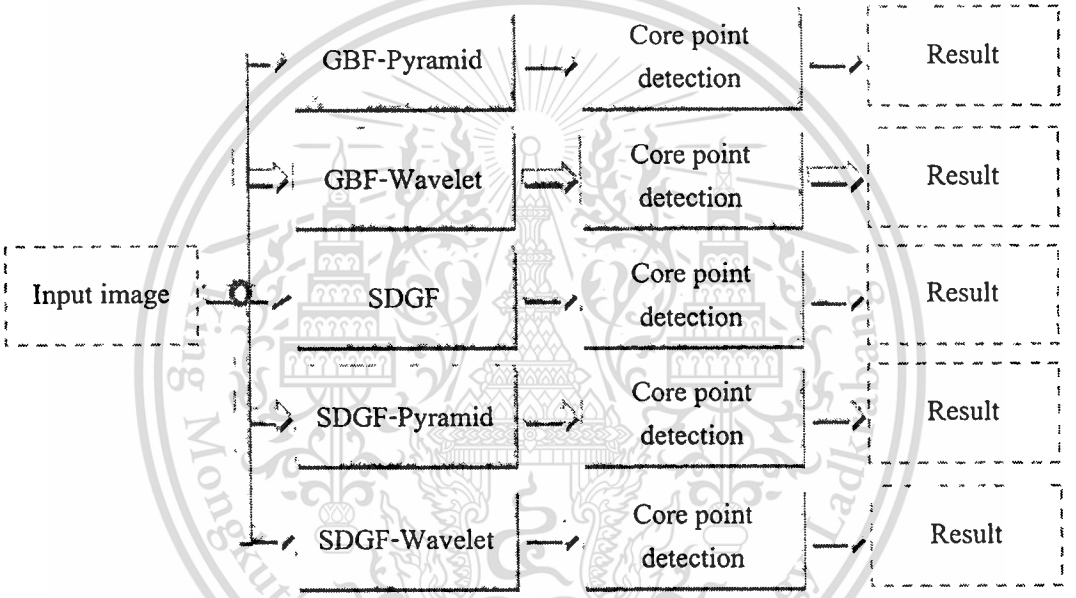


Figure 5.16 Data flow diagram of fingerprint core point detection our proposed techniques.

5.4.2 Result on core point detection

The results of those fingerprint images with 2 cores are shown in Figure 5.18, Figure 5.20 and Figure 5.22. We classified the obtained results into:

i) 2-core detected correctly (no alias at all),

ii) 2-core detected but together with some aliases,

iii) only one core can be detected and some aliases present,

iv) no core detected but only with aliases, and

v) no point detect at all (fail case).

Similarly we divided the results of those 1 core images (561 images) as shown in Figure 5.19, Figure 5.21 and Figure 5.23. Similar results were obtained. We can conclude that the correct detection of the core point could be improved significantly when our techniques is applied. With such an attempt, alias point detection could reduce. The filter has removed noise effectively. The ridge and valley are greatly enhanced, such that the core point detection method can work with less error. The experiments of core point detection we have divide into two parts: none and conventional Gabor filtering.

As illustrated in Figure 5.17 a) the core point detection results of the fingerprints image, none enhancement the algorithm has detected many wrong core points when the images are not filtered. However those alias points can be reduced dramatically when the same images are enhanced as shown in Figure 5.17. The conventional method corresponding to Gabor filtering (C-GBF) shown in Figure 5.17 b) and Figure 5.17 c) the Second derivative of Gaussian filter (SDGF) these alias can be reduced but ridges is no slightly. From Figure 5.17 d) GBF-Pyramid technique, Figure 5.17 e) SDGF-Pyramid technique, Figure 5.17 f) GBF-Directional wavelet transform and Figure 5.17 g) SDGF-Directional wavelet transform those are corresponding to proposed techniques . The enhancement results show that the filter preserves the continuity of the ridge flow pattern and enhances the clarity of the ridge and valley structures. In addition to reducing noise (alias) in the image, the filter is able to fill in small breaks that occur within ridge. The ridge and valley are greatly enhanced slightly, such that the core point detection method can work with less error.

a) Fingerprint none-enhancement

b) Conventional Gabor filter (C-GBF)

c) Second derivative of Gaussian filter (SDGF)

d) GBF-Pyramid technique

e) SDGF-Pyramid technique

f) GBF-Directional wavelet transform

g) SDGF-Directional wavelet transform

Figure 5.17 Core points detection results of various fingerprints image.

In the Table 5.1 and Table 5.2, we can summarized numbers of the correct detection of the core point could be improved significantly when each enhancement techniques are applied. With such an attempt, alias point detection could reduce. The filter has removed noise effectively. The ridge and valley are greatly enhanced slightly, such that the core point detection method can work with less error. The result is quite convincing. However the enhancement cannot provide dramatic improvement.

Table 5.1 Number of detection results of 2-core point fingerprint pattern (whorl). The total number of images is 185 images.

Detection / Enh. method	2-core det. (no alias)	2-core det. (with alias)	1-core det. (with alias)	Alias det. only	Fail
No Enhancement	74	70	29	8	4
C-GBF	98	64	14	6	3
SDGF	122	45	14	2	2
GBF-Pyramid	122	38	25	0	0
SDGF-Pyramid	139	21	24	0	1
GBF-Wavelet	150	18	15	1	1

None-Enh: none enhancement

C- GBF: conventional Gabor filtering

SDGF: second derivative of Gaussian filter

Table 5.2 Number of detection result of 1-core point fingerprint pattern (most are left-loop and right loop). The total number of images is 561 images.

Detection / Enh. method	1-core det. (no alias)	1-core det. (with alias)	Alias det. only	Fail
No Enhancement	187	272	74	28
C-GBF	317	186	32	26
SDGF	349	193	17	13
GBF-Pyramid	388	165	7	1
SDGF-Pyramid	380	166	7	8
GBF-Wavelet	415	131	10	5
SDGF-Wavelet	418	142	1	0

The purpose of a fingerprint enhancement technique is to improve the quality of input fingerprint images and make them more suitable for the minutia extraction in an automatic fingerprint identification system. Therefore, the final criterion of evaluating such an enhancement techniques is the amount of performance improvement when the techniques is applied to the noisy fingerprint images. From in the Table 5.1 and Table 5.2, the obtained results, in the first experiment, the fingerprint enhancement techniques was not applied (note None-Enh.) The percentages of accurate 2-core point detection shown in Figure 5.18, Figure 5.20 and Figure 5.22, and percentages accurate of 1-core point detection shown in Figure 5.19, Figure 5.21 and Figure 5.23 those percentages are less than our proposed techniques applied. In the second experiment, each proposed techniques are applied directly to fingerprint images. From these experimental results, we can observe that the performance of the fingerprint core point detection is significantly improved when our fingerprint enhancement techniques is applied to the input fingerprint images. In particular, the enhancement techniques substantially reduced the alias in occurs the core point detect while percentage of correct detection of the core point could be improved when proposed techniques are applied. With such an attempt, alias point detection could be dramatically reduced. We have recommendation for enhance techniques, the Gabor filter and Second derivative of Gaussian filter on the scheme of directional wavelet transform is better than the Gabor filter and Second derivative of Gaussian filter on the scheme of pyramid technique.

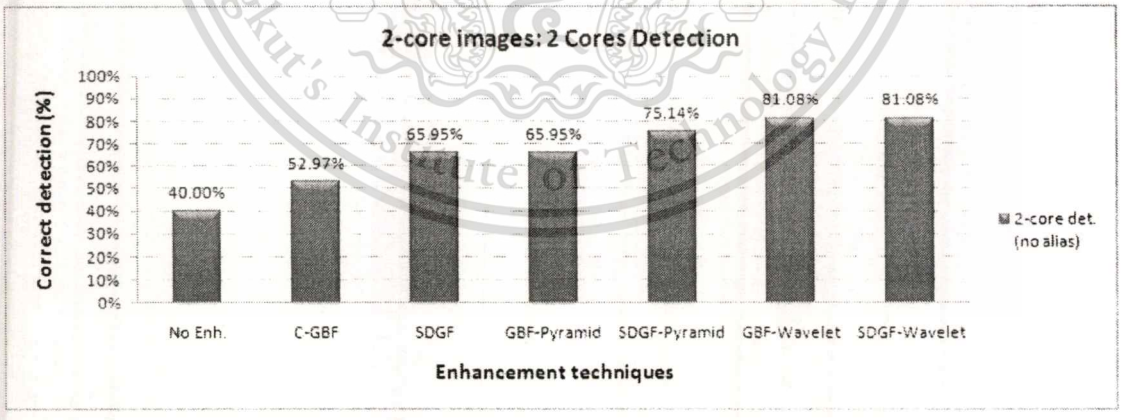


Figure 5.18 The correct detection results of 2-core point fingerprint patterns (whorl) according to several filtering techniques; 185 images in total.

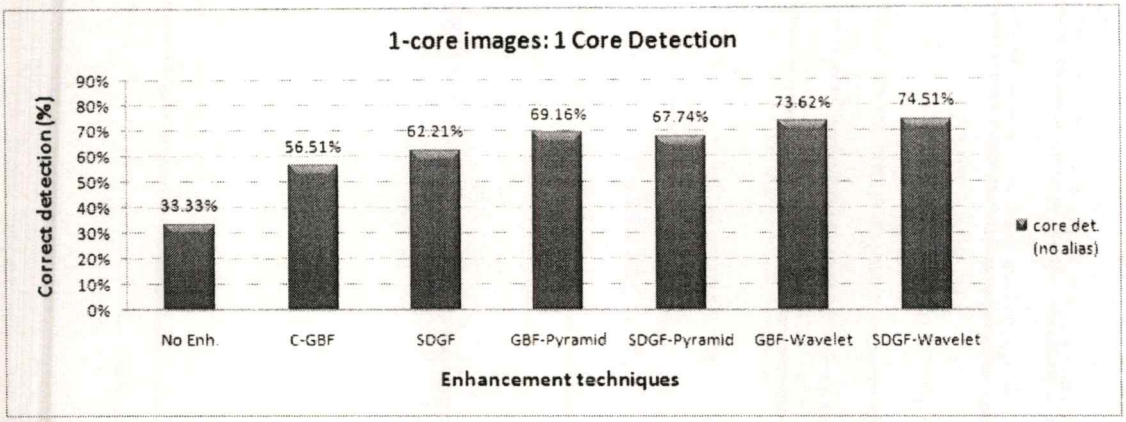


Figure 5.19 The correct detection result of 1-core point fingerprint patterns (most are left-loop and right-loop) according to several filtering techniques; 561 images in total.

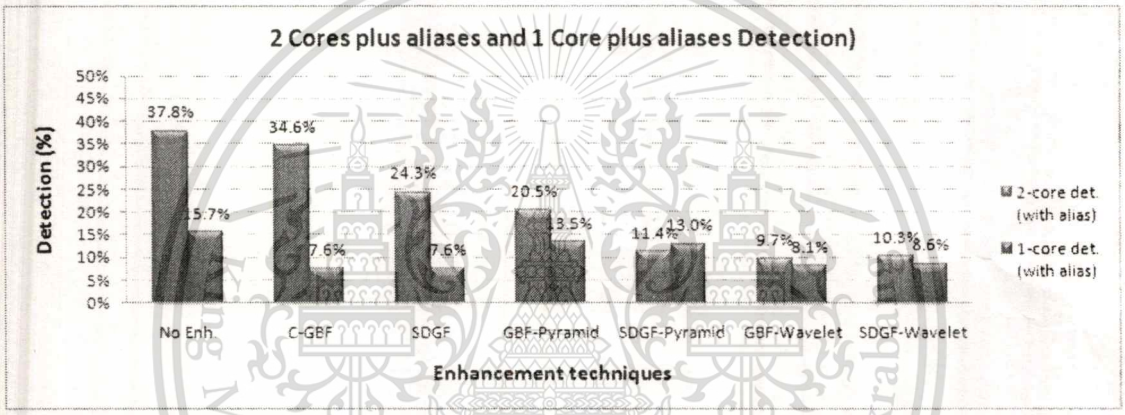


Figure 5.20 Fail detection results of 2-core point fingerprint patterns according to several filtering techniques.

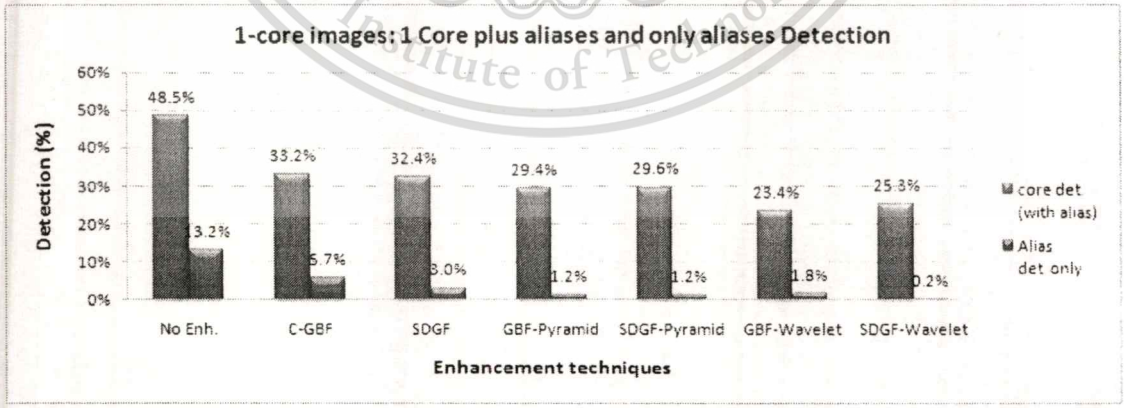


Figure 5.21 Fail detection result of 1-core point fingerprint patterns according to several filtering techniques.

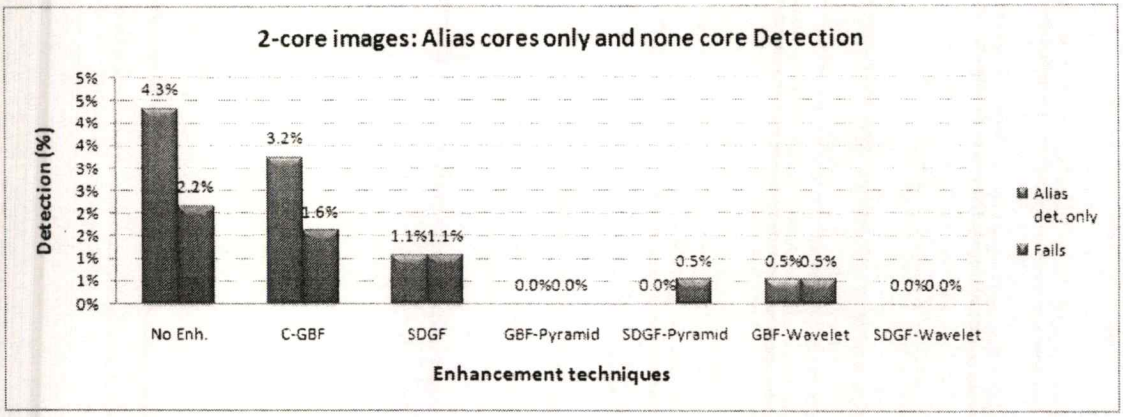


Figure 5.22 Fail detection results of 2-core point fingerprint patterns according to several filtering techniques.

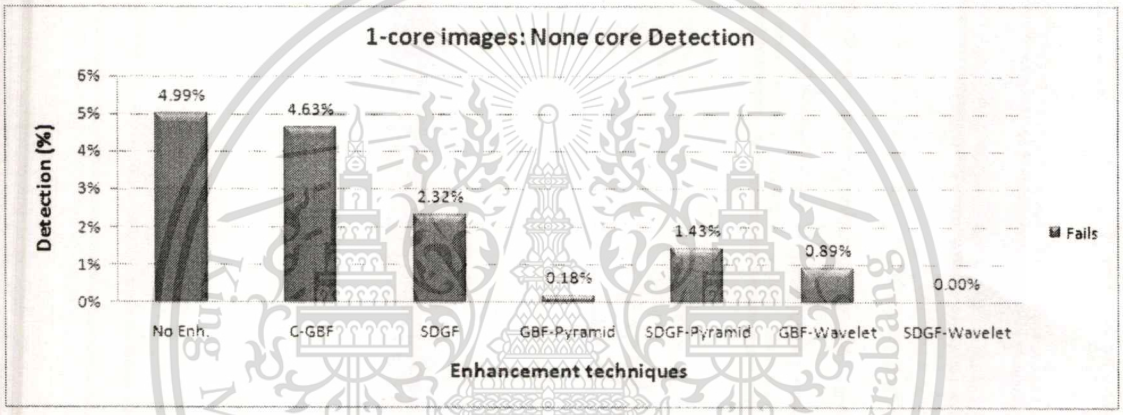


Figure 5.23 None core detection result of 1-core point fingerprint pattern according to several filtering techniques.

5.4.3 Result on minutiae

In this section, experiment results of minutiae extraction that we have randomly chosen some of fingerprint images (see Figure 5.25) for testing minutiae extraction all are quite varied in image quality. However, all holds core point. Orientation field estimation is applied after image normalization. Two type of directional filters; namely Gabor filter and Second derivative of Gaussian filter are investigated together with other two arrangements which are directional wavelet transform technique and pyramid technique. After a fingerprint image has been enhanced, the next step is to extract the minutiae from the enhanced image. The Crossing Number (CN) technique is able to extract the minutiae from the skeleton image. The obtained results are given in Table 5.3 and Table 5.2. Table 5.4 demonstrates the ridge end points whilst table 5.4

demonstrates the bifurcation. Shown in Figure 5.24 are the example images of various process steps.

a) Origianal

b) Thinning of c)

c) Gabor filter

d) Gabor-Wavelet

e) Gabor Pyramid

f) SDGF

g) SDGF-Wavelet

h) SDGF-Pyramid

Figure 5.24 Results of performing minutiae extraction on a fingerprint image at various process step.

Shown in Table 5.3 and Table 5.4, our counting is based on the genuine point. The positive or negative value means how big the difference the number of points it holds compare with the number of genuine points. It is quite clear that corporation of the filter with either wavelet transform or pyramid technique can offer better performance as the number of ridge end point decreases when compared to that of applying the filter only. However, one also can notice that the filter application can destroy the genuine end point. Such a situation is demonstrated by the negative difference. This can happen because the too-close ridges are joined. According to Table 5.3, Image #2 is greatly influenced by the said conditions.

Application of the filter for image enhancement seems to have less effect on bifurcation. In many cases, the destroyed line-ends are because of their joining and a bifurcation is generated. If one looks into Table 5.4, the said conditions are revealed. Image #2 (see also Figure 5.25) holds some thicker line near its end point the space between the considering line and the neighboring line is then lower. According to the field smoothing, those line are joined and a bifurcation is generated.

Table 5.3 Ridge end detection

		Images					
		#1	#2	#3	#4	#5	#6
	Number of Genuine Ridge End	13	9	5	13	23	8
After Enhancement	Gabor filter	+1	0	-1	+1	+1	0
	Gabor - Wavelet	0	-1	0	-1	+1	-1
	Gabor - Pyramid	-1	-1	+1	+1	0	+1
	SDGF	+1	-1	+1	+1	0	0
	SDGF - Wavelet	-1	-2	+1	+1	+1	0
	SDGF - Pyramid	-1	-1	0	+1	+1	+1

Table 5.4 Bifurcation detection

		Images					
		#1	#2	#3	#4	#5	#6
	Number of Genuine Bifurcate	8	13	13	16	13	16
After Enhancement	Gabor filter	0	-1	0	-1	+1	0
	Gabor - Wavelet	+1	0	0	0	-1	0
	Gabor - Pyramid	+1	0	-1	0	-1	0
	SDGF	-1	0	0	0	0	0
	SDGF - Wavelet	0	+2	0	-1	0	0
	SDGF - Pyramid	0	0	0	-1	-1	0

Figure 5.25 Some of fingerprint in experiment for minutiae.

5.5 Computational Cost

According to the enhancement; on one hand, we can have clearer images desired for further processing, on another hand we have to pay for the cost of each the enhancement techniques. The computation has been carried out with a Laptop computer with CPU 2.16 GHz and 3GHz of RAM. The fingerprint images that we used were captured with optical sensor image size is 640 x 480 pixels with 500 dpi resolutions. However, we did investigation by averaging 10 input images, we found that enhance procedure has taken times greater than without enhancement as shown in the Table 5.5. It should be noted that a Matlab package is used in this experiment. Reading and displaying the image can consume a certain time. This intrinsic time is therefore counted in without any enhancement routines.

In comparative term, rough figures are also given. Simple filtering takes 3 times larger. Wavelet and pyramid take 4 times and 7 times respectively. Intrinsic time is used as a base of this computation.

Table 5.5 Shown computational time processing with enhanced techniques.

Without Enh.	With Gabor filter Enh.	With SDGF Enh.	Pyramid with Gabor filter Enh.	Pyramid with SDGF. Enh.	Wavelet with Gabor filters Enh.	Wavelet with SDGF Enh.
3.94208 sec	11.00434 sec	11.46766 sec	29.62928 sec	26.87362 sec	15.90271 sec	15.73267 Sec
1x	3x	3x	7x	7x	4x	4x

Using our enhancement techniques, we could improve the correctness of core point detection. In many cases, as shown in Figure 5.29 below, the algorithm still went wrong. Most failures are because of the really poor quality of images (completely dark and/or completely dry in some image portions). We still need more procedure to eliminate these alias points.

Figure 5.26 Example of image patterns that cause algorithm failed.

5.6 Summary

According to the results of experiments given in this chapter, fairly clear that the application of directional filtering to the enhancement process has improved the image dramatically; no matter in term of its visualization or the success in core point detection. Application of only Gabor filter or the second derivative of Gaussian filter can improve the image for some certain degrees. Including such filters into more sophisticate schemes such as pyramid technique or directional wavelet transform has shown a good improvement of enhancement performance. Of course, seems to be in line with a matter of fact that we have to pay more in computing resource, i.e. longer computational time. Pyramid technique requires the largest computational time whist offering similar performance compared to DWT.

Application of directional filtering seems to have less effect to the consistency of line-end and bifurcation. This can be confirmed by the result given at very end of this chapter. We can see that less than 10% changes in the number of those 2 features can be estimated when the filter is applied. Therefore, directional filtering can be also nicely applied for minutiae extraction.

A major note can be taken here: the application of filter can improve the quality of the image but cannot repair a large area with completely black or white stain. Therefore some fake core points (and fake line-ends) still left unsolved. More steps may be required to remove these points if needed.

Chapter 6

Discussion, Conclusion & Future Prospects

6.1 Discussion

With the advancement of biometrics, due to increase in frauds and crimes, more and more biometric modalities are emerging. Fingerprints are widely used in many personal identification systems due to its permanence and uniqueness. Automatic fingerprint identification systems are being commercially deployed at many places. Hence, error rate of just one percentage may prove to very disastrous. Currently, many researchers are trying to develop systems with one hundred percentage recognition rate.

One of the very important factors which can affect the performance of a fingerprint identification system is the quality of the input images. Several factors determine the quality of fingerprint image: skin conditions (e.g. wetness, dryness, dirtiness, temporary or permanent cuts or bruises), sensor conditions (e.g. dirtiness, noise, size), user cooperation etc. The poor image quality leads to many pseudo-minutiae. This degrades the performance of fingerprint identification system. In a typical fingerprint identification system first of all acquisition takes place which means acquiring of fingerprint data from the user, then comes the feature extraction part which involves use of various image processing applications and finally matching is done with template feature sets in the database

A benefit of the proposed techniques is that they are possible to implement classification and automatic fingerprint identification system (recognition) applied directly to the original grayscale images, avoiding morphological operations all together. The latter also adds not the least to the accurateness of the minutia extraction. Finally, our fingerprint enhancement with directional could also provide added value (complementary information) in non-minutiae based fingerprint recognition systems and reinforce the minutia definition.

6.2 Conclusion

In this thesis, we purpose of fingerprint image enhancement with directional filtering that involves two types similar of filters: Gabor filter and Second derivative of Gaussian filter are also

applied in the scheme of directional wavelet transform and pyramid technique. The fingerprint is to improve the clarity of ridges and valleys of input fingerprint images and make them more suitable for the classification, automatic fingerprint identification and minutiae extraction systems. The ultimate criterion for evaluating such an enhancement technique is the total amount of quality core point detected improvement when the enhancement technique is applied to the noisy input fingerprint images. Examples of the enhancement results are shown in (chapter 5). From these examples, we can see that our enhancement algorithm does improve the clarity of the ridge and valley structures of input fingerprint images.

The experiments were conducted using FVC 2004 database of fingerprint images; all are quite varied in image quality. However, all holds core point. Orientation field estimation is applied after image normalization. Two type of directional filters; namely Gabor filter and Second derivative of Gaussian filter are investigated together with other two arrangements which are directional wavelet transform technique and pyramid technique. The experiment results by using a combination of both fingerprints without and with enhancement in accurate of core point detection. Each enhancement techniques are able to effectively enhance the clarity of the ridge and valley structures while reducing noise. In contrast, for low quality images that exhibit high intensities of noise, the filter is less effective in enhancing the image due to inaccurate of the core point detection. However, in practice, this does not pose a significant limitation as fingerprint matching techniques generally place more emphasis on the well-defined regions, and will disregard an image if it is severely corrupted. Overall, the results have shown that the implemented enhancement techniques are a useful step to employ prior to minutiae extraction.

The effectively of performance fingerprint images enhancement base minutiae extracted by the use of directional filtering. Two similar filters: Gabor filter and second derivative of Gaussian filter are investigated. The corporation of these two filters in Directional wavelet arrangement and pyramid technique enhancement are also involved. Despite the computation complexity, the extra arrangement of the filter can perform the broken line connection. However, another obvious disadvantage is that: it does wrong join the too-close ridge. As a result, it sometimes produces some fake bifurcations. For a poor quality image and even an expert do the examination, it is really difficult to justify for the interest feature. In many cases, the broken lines still cannot be joined by the filter. The fake ridge ends still exist. The window size used during the field estimation process is also important in such a way that; on one side too big window can give

the good field smoothing but it can ignore the bifurcation. On the other side the too small window size cannot join well the broken ridge but it pick up well those fake bifurcations. We conclude our observation that designing of the filter is fairly critical. Its performance can be good for some type of poor images but may not so efficient for others. In most cases directional filter can remove noise dramatically. The Crossing Number method was implemented to perform extraction of minutiae. Experiments conducted have shown that this method is able to accurately detect all valid bifurcations and ridge endings from the thinned image. Overall, I have implemented a set of reliable techniques for fingerprint image enhancement. These techniques can then be used to facilitate the further study of the statistics of fingerprints. In addition, these techniques can be also applied in other fingerprinting applications such as fingerprint matching and classification.
The results support that even state-of-the-art fingerprint recognition systems benefit from the proposed techniques, especially the ones that make use accurate core point location.

6.3 Future Prospects

Biometrics is growing at a very fast rate and researchers all over the world are trying to achieve the target of one hundred percentages recognition rate. A biometric system consists of different algorithms and if improving any algorithm can help in achieving the target, it will be a stepping stone to future works. An investigation into a filter whose primary aim is to specifically enhanced both the minutia points and singular points.

This thesis has followed the approach adopted by most previous work where the emphasis is on enhancing the ridge structures using directional filtering. However, while the ridge structures are enhanced, this approach has shown to be less effective in enhancing areas with no information at all (i.e. completely dark and/or completely white). At the rim of such region there also can appear fake line-ends or even fake core (and/or delta) points that have to be removed by other algorithms. Rather than fingerprint images we also hope the directional filters together with either pyramid or DWT can also be applied to other area applications.

The clearer fingerprint image offered by the enhancement techniques proposed in this thesis can be of course utilized by either fingerprint classification and/or matching processes which are also more rooms for researcher to exploit the higher performance ones compared to the currently exist algorithms.

References

[1] D. Maltoni, D. Maio, A.K. Jain, S. Prabhakar , Second Editor "Handbook of Fingerprint Recognition" Springer, London, 2009

[2] Kamei T., "Image filter design for fingerprint enhancement," in *Automatic Fingerprint Recognition Systems*, Springer, New York, pp. 113–126, 2

[3] L. Hong, A. K. Jain, S. Pankanti, and R. Bolle, "Fingerprint Enhancement", 3rd IEEE Workshop on Applications of Computer Vision (WACV '96) p. 202.

[4] Kamei, T. and M. Mizoguchi, Image filter design for fingerprint enhancement, *Proc. ISCV 95*, Coral Gables, FL, 1995, pp. 109–114.

[5] **Mayank Vatsa , Richa Singh,** "Biometric Technologies", *Indian Institute of Technology Kanpur and Electronic , India*

[6] Gonzales R.C. and Woods R.E., *Digital Image Processing*, 3rd edition, Prentice-Hall, Englewood Cliffs, NJ, 2007.

[7] Ito K., Morita A., Aoki T., Nakajima H., Kobayashi K. and Higuchi T., "A Fingerprint Recognition Algorithm Combining Phase-Based Image Matching and Feature-Based Matching," in *Proc. Int. Conf. on Biometrics*, LNCS 3832, pp. 316–325, 2006.

[8] Coetzee L. and Botha E.C., "Fingerprint recognition in low quality images," *Pattern Recognition*, vol. 26, no. 10, pp. 1441–1460, 1993.

[9] Lindoso A., Entrena L., Liu-Jimenez J. and San Millan E., "Correlation-Based Fingerprint Matching with Orientation Field Alignment," in *Proc. Int. Conf. on Biometrics*, LNCS 4642, pp. 713–721, 2007.

[10] Sujan V.A. and Mulqueen M.P., "Fingerprint identification using space invariant transforms," *Pattern Recognition Letters*, vol. 23, no. 5, pp. 609–619, 2002.

[11] Ouyang Z., Feng J., Su F. and Cai A., "Fingerprint Matching with Rotation-Descriptor Texture Features," in *Proc. Int. Conf. on Pattern Recognition (18th)*, vol. 4, pp. 417–420, 2006

[12] Wilson C.L., Watson C.I. and Paek E.G., "Combined optical and neural network fingerprint matching," *Proc. of SPIE (Optical Pattern Recognition VIII)*, vol. 3073, pp. 373–382, 1997.

[13] Kryszczuk K.M., Morier P. and Drygajlo A., "Study of the Distinctiveness of Level 2 and Level 3 Features in Fragmentary Fingerprint Comparison," in *Proc. Workshop on Biometric Authentication (in ECCV* 2004*)*, LNCS 3087, pp. 124–133, 2004.

[14] Chen J. and Moon Y.S., "A Minutiae-Based Fingerprint Individuality Model," in *Proc. Conf. Computer Vision and Pattern Recognition*, 2007.

[15] Jea T.Y. and Govindaraju V., "A minutia-based partial fingerprint recognition system," *Pattern Recognition*, vol. 38, no. 10, pp. 1672–1684, 2005.

[16] Bishnu A., Das S., Nandy S.C. and Bhattacharya B.B., "Simple algorithms for partial point set pattern matching under rigid motion," *Pattern Recognition*, vol. 39, no. 9, pp. 1662–1671, 2006.

[17] Ballard D.H., "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognition*, vol. 3, no. 2, pp. 110–122, 1981.

[18] Ranade A. and Rosenfeld A., "Point pattern matching by relaxation," *Pattern Recognition*, vol. 12, no. 2, pp. 269–275, 1993.

[19] Baird H., *Model Based Image Matching Using Location*, MIT Press, Cambridge, MA,1984.

[20] Oh C. and Ryu Y.K., "Study on the center of rotation method based on minimum spanning tree matching algorithm for fingerprint recognition," *Optical Engineering*, vol. 43, no. 4, pp. 822–829, 2004.

[21] Tan X. and Bhanu B., "Fingerprint matching by genetic algorithms," *Pattern Recognition*, vol. 39, no. 3, pp. 465–477, 2006.

[22] Bazen A.M. and Gerez S.H., "Systematic methods for the computation of the directional fields and singular points of fingerprints," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 24, no. 7, pp. 905–919, 2002.

[23] Jain A.K., Hong L. and Bolle R., "On-line fingerprint verification," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 19, no. 4, pp. 302–313, 1997.

[24] Bazen A.M. and Gerez S.H., "An Intrinsic Coordinate System for Fingerprint Matching," in *Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (3rd)*, pp. 198–204, 2001a.

[25] Cheng J., Tian J. and Chen H., "Fingerprint Minutiae Matching with Orientation and Ridge," in *Proc. Int. Conf. on Biometric Authentication (1st)*, LNCS 3072, pp. 351– 358, 2004.

[26] Yager N. and Amin A., "Evaluation of Fingerprint Orientation Field Registration Algorithms," in *Proc. Int. Conf. on Pattern Recognition (17th)*, vol. 4, pp. 641–644, 2004.

[27] Gu J., Zhou J. and Yang C., "Fingerprint recognition by combining global structure and local cues," *IEEE Transactions on Image Processing*, vol. 15, no. 7, pp. 1952–1964, 2006.

[28] Wan D. and Zhou J., "Fingerprint recognition using model-based density map," *IEEE Transactions on Image Processing*, vol. 15, no. 6, pp. 1690–1696, 2006.

[29] Jain et al. (2000). Jain A.K., Prabhakar S., Hong L. and Pankanti S., "Filterbank-based fingerprint matching," *IEEE Transactions on Image Processing*, vol. 9, pp. 846–859, 2000.

[30] Chen, Tian and Yang (2006). Chen X., Tian J. and Yang X., "A new algorithm for distorted fingerprints matching based on normalized fuzzy similarity measure," *IEEE Transactions on Image Processing*, vol. 15, no. 3, pp. 767–776, 2006.

[31] Xie, Su and Cai (2006). Xie X., Su F. and Cai A., "Ridge-Based Fingerprint Recognition," in *Proc. Int. Conf. on Biometrics*, LNCS 3832, pp. 273–279, 2006.

[32] Feng, Ouyang and Cai (2006). Feng J., Ouyang Z. and Cai A., "Fingerprint matching using ridges," *Pattern Recognition*, vol. 39, no. 11, pp. 2131–2140, 2006.

[33] Feng and Cai (2006a). Feng J. and Cai A., "Fingerprint Representation and Matching in Ridge Coordinate System," in *Proc. Int. Conf. on Pattern Recognition (18th)*, vol. 4, pp. 485–488, 2006a.

[34] Stosz and Alyea (1994). Stosz J.D. and Alyea L.A., "Automated System for Fingerprint Authentication Using Pores and Ridge Structure," in *Proc. of SPIE (Automatic Systems for the Identification and Inspection of Humans)*, vol. 2277, pp. 210–223, 1994.

[35] Kryszczuk, Morier and Drygajlo (2004). Kryszczuk K.M., Morier P. and Drygajlo A., "Study of the Distinctiveness of Level 2 and Level 3 Features in Fragmentary Fingerprint Comparison,"

[36] Jain, Chen and Demirkus (2007). Jain A.K., Chen Y. and Demirkus M., "Pores and ridges: Highresolution fingerprint matching using Level 3 features," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 29, no. 1, pp. 15–27, 2007.

[37] L. Hong and A. K. Jain, Classification of fingerprint images, *MSU Technical Report*, MSU Technical Report MSUCPS:TR98-18, June 1998.

[38] Kawagoe M. and Tojo A., "Fingerprint pattern classification," *Pattern Recognition*, vol. 17, pp. 295–303, 1984.

[39] Fu K.S. and Booth T.L., "Grammatical inference: Introduction and survey: Part I," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 8, no. 3, pp. 343–360, 1986a.

[40] Moayer B. and Fu K., "A tree system approach for fingerprint pattern recognition," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 8, no. 3, pp. 376– 388, 1986.

[41] Rao K. and Balck K., "Type classification of fingerprints: A syntactic approach," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 2, no. 3, pp. 223–231, 1980.

[42] Bunke H., "Structural and syntactic pattern recognition," in *Handbook of Pattern Recognition & Computer Vision*, C.H. Chen et al. (Eds.), World Scientific, Singapore, 1993.

[43] Maio D. and Maltoni D., "A Structural Approach to Fingerprint Classification," in *Proc. Int. Conf. on Pattern Recognition (13th)*, 1996.

[44] Yao Y., Marcialis G.L., Pontil M., Frasconi P. and Roli F., "Combining flat and structured representations for fingerprint classification with recursive neural networks and support vector machines," *Pattern Recognition*, vol. 36, no. 2, pp. 397–406, 2003.

[45] Neuhaus M. and Bunke H., "A Graph Matching Based Approach to Fingerprint Classification Using Directional Variance," in *Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (5th)*, pp. 191–200, 2005.

[46] Jain A.K., Duin P.W. and Mao J., "Statistical pattern recognition: A review," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 22, no. 1, pp. 4–37, 2000.

[47] Tan X., Bhanu B. and Lin Y., "Fingerprint classification based on learned features," *IEEE Transaction on Systems, Man, and Cybernetics, Part C*, vol. 35, no. 3, pp. 287–300, 2005.

[48] Hong J.H., Min J.K., Cho U.K. and Cho S.B., "Fingerprint classification using one-vs-all support vector machines dynamically ordered with naive Bayes classifiers," *Pattern Recognition*, vol. 41, no. 2, pp. 662–671, 2008.

[49] Fitz A.P. and Green R.J., "Fingerprint classification using hexagonal fast Fourier transform," *Pattern Recognition*, vol. 29, no. 10, pp. 1587–1597, 1996.

[50] Jain A.K., Prabhakar S. and Hong L., "A multichannel approach to fingerprint classification," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 21, no. 4, pp. 348–359, 1999.

[51] Li J., Yau W.Y. and Wang H., "Combining singular points and orientation image information for fingerprint classification," *Pattern Recognition*, vol. 41, no. 1, pp. 353–366, 2008.

[52] Cappelli R., Lumini A., Maio D. and Maltoni D., "Fingerprint classification by directional image partitioning," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 21, no. 5, pp. 402–421, 1999.

[53] Hughes P. and Green A., "The Use of Neural Networks for Fingerprint Classification," in *Proc. Int. Conf. on Neural Networks (2nd)*, 1991.

[54] Bowen J., "The Home Office Automatic Fingerprint Pattern Classification Project," in *Proc. IEE Colloquium on Neural Networks for Image processing Applications*, 1992.

[55] Wilson C.L., Candela G.T. and Watson C.I., "Neural network fingerprint classification," *Journal of Artificial Neural Networks*, vol. 1, no. 2, pp. 203–228, 1994.

[56] Candela G.T., Grother P.J., Watson C.I., Wilkinson R.A. and Wilson C.L., "PCASYS – A Pattern-Level Classification Automation System for Fingerprints," Tech. Report: NIST TR 5647, Aug. 1995.

[57] L. Hong, Y. Wan, and A. Jain. "Fingerprint image enhancement: Algorithm and performance evaluation," IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(8):777–789, August 1998.

[58] Sherlock B.G., Monro D.M. and Millard K., "Fingerprint enhancement by directional Fourier filtering," *IEE Proceedings Vision Image and Signal Processing*, vol. 141, no. 2, pp. 87–94, 1994

[59] Wang W., Li J., Huang F. and Feng H., "Design and implementation of Log-Gabor filter in fingerprint image enhancement," *Pattern Recognition Letters*, vol. 29, no. 3, pp. 301–308, 2008

[60] Chikkerur S., Cartwright A.N. and Govindaraju V., "Fingerprint enhancement using STFT analysis," *Pattern Recognition*, vol. 40, no. 1, pp. 198–211, 2007

[61] Jiang X., "A Study of Fingerprint Image Filtering," in *Proc. Int. Conf. on Image Processing*, 2001

[62] Kamei T. and Mizoguchi M., "Image Filter Design for Fingerprint Enhancement," in *Proc. Int. Symp. on Computer Vision*, pp. 109–114, 1995

[63] Hong L., Jain A.K., Pankanti S. and Bolle R., "Fingerprint Enhancement," in *Proc. Workshop on Applications of Computer Vision*, pp. 202–207, 1996

[64] Bernard S., Boujemaa N., Vitale D. and Bricot C., "Fingerprint Segmentation Using the Phase of Multiscale Gabor Wavelets," in *Proc. Asian Conf. Computer Vision*, 2002

[65] Almansa A. and Lindeberg T., "Fingerprint enhancement by shape adaptation of scale-space operators with automatic scale selection," *IEEE Transactions on Image Processing*, vol. 9, no. 12, pp. 2027–2042, 2000

[66] Hsieh C.T., Lai E. and Wang Y.C., "An effective algorithm for fingerprint image enhancement based on wavelet transform," *Pattern Recognition*, vol. 36, no. 2, pp. 303–312, 2003

[67] Cheng J. and Tian J., "Fingerprint enhancement with dyadic scale-space," *Pattern Recognition Letters*, vol. 25, no. 11, pp. 1273–1284, 2004

[68] Fronthaler H., Kollreider K. and Bigun J., "Pyramid-Based Image Enhancement of Fingerprints," in *Proc. Workshop on Automatic Identification Advanced Technologies*, pp. 45–50, 2007

[69] Fronthaler H., Kollreider K. and Bigun J., "Local features for enhancement and minutiae extraction in fingerprints," *IEEE Transactions on Image Processing*, vol. 17, no. 3, pp. 354–363, 2008

[70] Alasdair McAndrew , "An Introduction to Digital Image Processing with Matlab", Notes for SCM2511 Image Processing 1, Semester 1, 2004, School of Computer Science and Mathematics, Victoria University of Technology

[71] Michel Misiti ,Yves Misiti, Georges Oppenheim, Jean-Michel Poggi, "Wavelets and their Applications", *published in France by Hermes Science/Lavoisier in 2003, First published in Great Britain and the United States in 2007 by ISTE Ltd*

[72] Michel Misiti,Yves Misiti, Georges Oppenheim, Jean-Michel Poggi, "Wavelets and their Applications"

[73] Zhengmao Ye, Habib Mohamadian, and Yongmao Ye, "Information Measures for Biometric Identification via 2D Discrete Wavelet Transform," *Proceedings of the 3rd Annual IEEE Conference on Automation Science and Engineering Scottsdale, AZ, USA, Sept 22-25, 2007.*

[74] Miao-li WEN, Yan LIANG, Quan PAN, Hong-cai ZHANG, "A Gabor filter based fingerprint enhancement algorithm in wavelet domain". *Communications and Information Technology, 2005. ISCIT 2005. IEEE International Symposium on* 12-14 Oct. 2005. Volume 2, On page(s): 1468- 1471.

[75] A. Farina, Z.M. Kovacs-Vajna, and A. Leone, "Fingerprint minutiae extraction from skeletonized binary images," *Pattern Recognition, 32(5):877–889, 1999.*

[76] D. Maio, D. Maltoni, "Direct Gray-Scale Minutiae Detection in Fingerprints," *IEEE Trans. Pattern Anal. Machine Intell., Vol. 19, No. 1, pp. 27-40, 1997.*

[77] Raymond Thai "Fingerprint Image Enhancement and Minutiae Extraction", *The University of Western Australia, 2003*

[78] Feng Yue, Wangmeng Zuo, Kuanquan Wang "A Performance Evaluation of Filter Design and Coding Schemes for Palmprint Recognition", *19th International Conference on Pattern Recognition, ICPR 20008, 8-11 Dec, 2008*

[79] P. J. Burt and E. H. Shashua, "The laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. Com-31 No. 4, April, 1983.

[80] E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden. Pyramid Methods in Image Processing. RCA Engineer, 29(6):33-41, 1984.

[81] A. K. Jain, S. Prabhakar and L. Hong, "A Multichannel Approach to Fingerprint Classification," IEEE Trans. On PAMI, Vol.21, No.4, pp. 348-359, April 1999

[82] Sen Wang, Wei Wei Zhang ,Yang Sheng Wang, "Fingerprint Classification by Directional Fields", Proc. 4th IEEE Int. Conf. on Multimodal Interfaces (ICMI'02), 2002.

[83] A. Julasayvake and S. Choomchuay, "An Algorithm for Fingerprint Core Point detection," Proc. Of Int. Symp. On Signal Processing and its Applications 2007 (ISSPA-2007), United Arab Emirates, February 2007.

[84] Grasselli (1969). Grasselli A., "On the automatic classification of fingerprints," in *Methodologies of Pattern Recognition*, S. Watanabe (Ed.), Academic, New York, 1969.

[85] Kass and Witkin (1987). Kass M. and Witkin A., "Analyzing oriented patterns," *Computer Vision Graphics and Image Processing*, vol. 37, no. 3, pp. 362–385, 1987

[86] Ratha, N., Chen, S., and Jain, A. Adaptive flow orientation based feature extraction in fingerprint images. *Pattern Recognition 28*, 11 (1995), 1657–1672.

Appendix A

Biography

Personal Information

Name & Surname:	Mr. Keokanlaya Sihalath
Nationality:	Lao
Birth of date:	December 15, 1977
Place of birth:	Khammouane province, Lao P.D.R

Education

High diploma

Field:	Electronics Engineering
Duration:	1996 - 1999
Institute:	Department of Electronics, Faculty of Engineering, National University of Laos

Bachelor degree (IT Bridging course)

Field:	Information technology
Duration:	2002 - 2004
Institute:	Department of Electronics, Faculty of Engineering, National University of Laos

Master degree

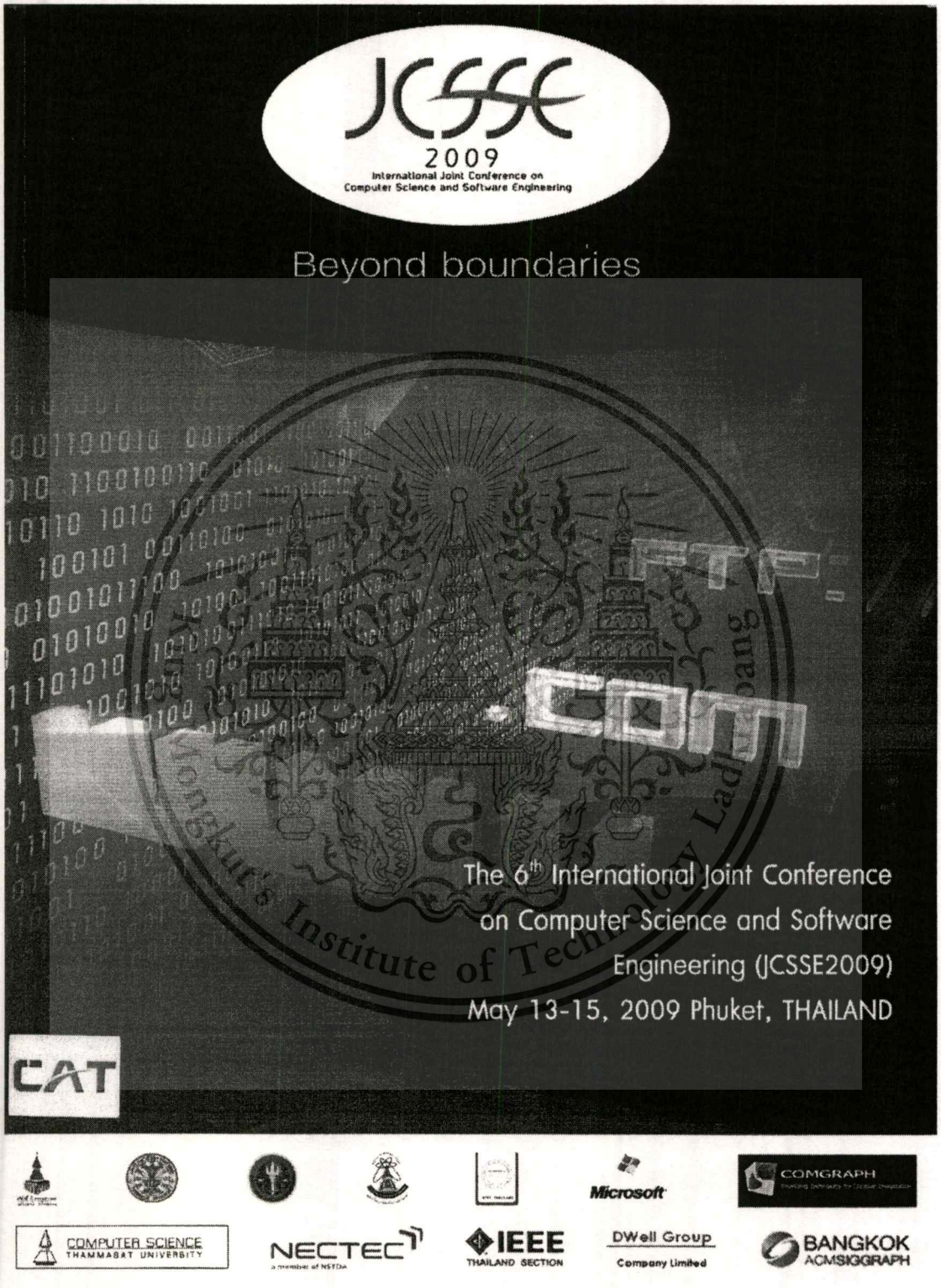
Field:	Electronics Engineering
Duration:	2008 - 2010
Institute:	Department of Electronics, Faculty of Engineering, King Mongkut's Institute of Technology, Ladkrabang, Thailand

Research Interests

Biometrics, Fingerprint recognition, Fingerprint image enhancement, Fingerprint classification and identification/verification

List of International Conference and Proceeding Papers

1. Keokanlaya Sihalath, Somsak Choomchuay, and Kazuhiko Hamamoto, "Core point Identification with Local Enhancement", *6th International Joint Conference on Computer Science and software Engineering on Computer Science and software Eng*ineering, May 13-15, 2009, Vol.1, Phutket, Thailand
2. Keokanlaya Sihalath, Somsak Choomchuay, Shatoshi Wada, and Kazuhiko Hamamoto, "PERFORMANCE EVALUATION OF FIELD SMOOTHING FILTERS" , *The 2nd Biomedical Engineering International Conference (BMEiCON 2009),* August 13-14, 2009, Phuket, Thailand
3. Keokanlaya Sihalath, Somsak Choomchuay, and Kazuhiko Hamamoto, "Directional Filtered Fingerprint Images" , *The First International Conference on Green Computing and 2nd AUN/SEED-Net Regional Conference on ICT*, 2-3 Mach 2010, GAD JAH MADA University, Indonesia
4. Keokanlaya Sihalath, Somsak Choomchuay, Shatoshi Wada, and Kazuhiko Hamamoto, "Fingerprint Image Enhancement with Second Derivative Gaussian Filter and Directional Wavelet Transform", *The 2nd International Conference on Computer Engineering and Applications* (ICCEA2010) 19-12 Mach 2010, Bali, Indonesia. Session3:Telecom Technology and Application,



Core point Identification with Local Enhancement

Keokanlaya Sihalath*, Somsak Choomchuay*, and Kazuhiko Hamamoto**

* Department of Electronics, Faculty of Engineering,
King Mongkut's Institute of Technology Ladkrabang (KMITL), Bangkok, Thailand
Tel: +66-2326-4222, Fax +66-2-739-2398, E-mail: keokanlaya@gmail.com, kchsomsa@kmitl.ac.th

**Department of Information Media Technology, School of Information Technology and Electronics, Tokai University, Tokyo, Japan, Email: hama@keyeki.cc.u-tokai.ac.jp

Abstract

Core point detection is an important task in many automatic fingerprint identification systems. There already exists many core point detected algorithms. Almost of them can efficiently detect the core point when the fingerprint image is at quite high quality, but when the fingerprint image is of poor quality, the efficient of the algorithm degrades. In this paper, we propose core point detecting method based on local oriented enhanced image. We have evaluated the quality of the invoked local enhancement by means of the subsequent core point locating performance. The obtained results are positive and convincing for the purpose of more accurate core point detection.

Key Word: Fingerprint, enhancement, Gabor filter, core point detection

1. Introduction

Fingerprint-base identification has been use for a very long time. It is also one of the most important biometric technologies. A fingerprint is the pattern of ridges and valleys on the surface of a fingerprint [1]. The core point has played important roles in most fingerprints identify techniques. The success of the correct detection is very much relied on the image quality. In many cases, fingerprint with numerous discontinuous ridges (dry, wet, damped, scars and smudges) can cause errors in fingerprint identification process [2].

The goal of an enhancement algorithm is to improve the clarity of ridge structures of fingerprint images in recoverable regions and to remove the unrecoverable regions. The process is employed in order to improve the clarity of ridge structures of input fingerprint. Fingerprint enhancement can be applied to either a gray-level fingerprint image or a binary one. As ridges and valleys in a fingerprint image alternate and run parallel to each other in a local neighborhood. Spurious ridge structure may change the individuality of input fingerprints. Ridges and valleys in a local neighborhood form a sinusoidal-shaped plane wave, which has a well-defined frequency and orientation. There appear many algorithms and techniques proposed and applied to fingerprint image enhancement. In particular case, Hong et al, proposed a decomposition method to estimate the orientation field from a set of filtered images obtained by applying a bank of Gabor filters on the input fingerprint images [3,4].

In this paper, we propose an alternative core point detect method where the local enhancement is needed in prior. In section 2, we review fingerprint enhancement steps as follows: normalization, local orientation field estimation, local ridge frequency and direction filter. In section 3, we review an algorithm used for core point detection, a Poincare technique. In section 4, we compare the obtained results indicating the performance of applying the local enhancement, and finally we concluded the paper in section 5.

2. Local enhancement

With an original fingerprint image there are noises, and may be low quality such as dry, wet, damped, scars, smudges and so on. In most case, the degradation occurs in part. Without any attempts, it is hard to identify the core point and minutiae of such an image with those degradations. However, at this state noise could be removed by filtering techniques. Among those, directional filter is a promising one.

2.1 Algorithm

As shown in Fig. 1, the directional filtering together with ridge frequency estimation are inserted to the conventional pre-processing steps which involve normalization, local field orientation estimation, and segmentation.

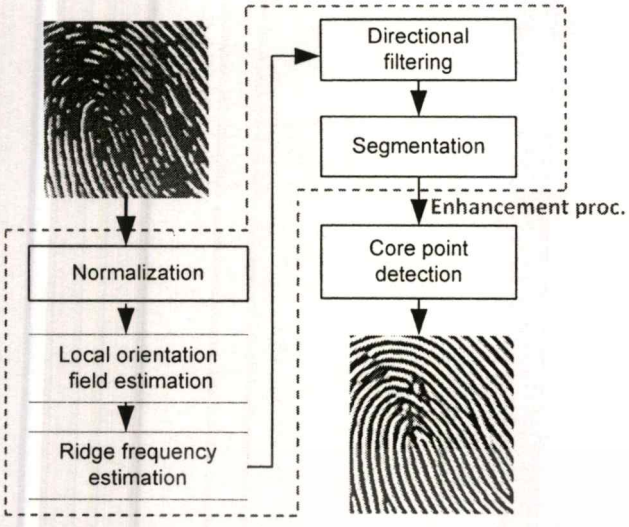


Figure 1: Experiment flow diagram

1. *Normalization*: An input fingerprint image is normalizing so that it has a pre-specified mean and variance.
2. *Local orientation field estimation*: The orientation image is estimate from the normalized input fingerprint image.
3. *Local ridge frequency estimation*: The ridge frequency estimation is applied after the normalization and local field estimation.
4. *Directional Filtering*: A bank of *Gabor filters* tuned to local ridge orientation and ridge frequency is applied to the ridge and valley pixels of the normalized image.
5. *Segmentation*: The segmentation is applied to crop the area of interest.

2.2 Normalization

Let, *I(i, j)* denote the gray-level value at pixel (*i, j*), *M* and *VAR* denote the estimated mean and variance of image *I*, respectively, and *N(i, j)* denote the normalized gray-level value at pixel (*i, j*). The normalized image is defined as: [1, 4]:

$$N(i,j)=\begin{cases} M_0+\sqrt{\dfrac{VAR_0(I(i,j)-M)^2}{VAR}} & \text{if } I(i,j)>M \\ M_0-\sqrt{\dfrac{VAR_0(I(i,j)-M)^2}{VAR}} & \text{otherwise} \end{cases} \quad (1)$$

where, M_0 and VAR_0 are the desired mean and variance values, respectively. In this work, we set values mean $M_0=0.5$ and variance $VAR_0=1$.

2.3 Local orientation field estimation

Let $\theta(i,j)$ be defined as the *orientation field* of a finger print image. $\theta(i,j)$ represents the local ridge at pixel (i,j). Local ridge, however, is usually specified for a block rather than that of every pixel. Thus, an image is divided in to a set of non-overlapping blocks, size of $w\times w$. Each bock holds a single orientation. A procedure proposed for orientation estimation is summarized below.

1. Divide the input image *I* into consecutive (non-overlapping) blocks with size $w\times w$.

2. Compute the gradients $\partial_x(i,j)$ and $\partial_y(i,j)$ at each pixel (i,j) which is the center of the block. The gradient operator can be chosen according to the computational complexity.

3. Estimate the local orientation of each block centered at pixel *(i,j)* using the following equations.

$$V_x(i,j)=\sum_{u=i-w/2}^{i+w/2}\sum_{v=j-w/2}^{j+w/2}2\partial_x(u,v)\partial_y(u,v), \quad (2)$$

$$V_y(i,j)=\sum_{u=i-w/2}^{i+w/2}\sum_{v=j-w/2}^{j+w/2}\partial_x^2(u,v)\partial_y^2(u,v), \quad (3)$$

Subsequently,

$$\theta(i,j)=\tfrac{1}{2}\tan^{-1}\left(V_y(i,j)/V_x(i,j)\right) \quad (4)$$

Where, $\theta(i,j)$ is the least square estimate of the local ridge orientation of the block centered at pixel (i,j).

4. Assumed that the local ridge orientation varies slowly in a local neighborhood where no core point appears. The discontinuity of ridge and valley due to noise can be softening by applying a low pass filter. However, to apply a low pass filter the orientation image must be converted to a continuous vector field, witch defined as bellows:

$$\Phi_x(i,j)=\cos[2\theta(i,j)], \text{ and } \Phi_y(i,j)=\sin[2\theta(i,j)] \quad (5)$$

where Φ_x and Φ_y are x and y components the vector field, respectively. With the resulted vector field, the low-pass filters can be performed as bellows:

$$\Phi'_x(i,j)=\sum_{u=-w_\Phi/2}^{w_\Phi/2}\sum_{v=-w_\Phi/2}^{w_\Phi/2}G(u,v)\cdot\Phi_x(i-uw,j-vw), \quad (6)$$

$$\Phi'_y(i,j)=\sum_{u=-w_\Phi/2}^{w_\Phi/2}\sum_{v=-w_\Phi/2}^{w_\Phi/2}G(u,v)\cdot\Phi_y(i-uw,j-vw), \quad (7)$$

here G is a two dimension low-pass filter with unit integral and $w_\Phi\times w_\Phi$ specified a size of filter is (3×3).

5. Compute the smoothed orientation field (local ridge orientation at *(i,j)*) can then be computed as follows:

$$\theta'(i,j)=\frac{1}{2}\tan^{-1}\left(\Phi'_y(i,j)/\Phi'_x(i,j)\right) \quad (8)$$

2.4 Local ridge frequency estimation

In the gray-level fingerprint image, along ridge and valley can be model as a sinusoidal-shaped wave along the direction perpendicular to the local ridge orientation see Fig.2. Let $N(i,j)$ be the normalized image and $\theta'(i,j)$ be the smoothed orientation field image, and then the steps involved in local ridge frequency estimation are as follows [4].

1. Divide $N(i,j)$ image into blocks of size $w \times w$, 5×5 for which center at pixel (i,j).

2. Compute an oriented window of size $l \times w$ (32×5), for each block centered at the pixel (i,j); that is defined in the ridge coordinate system (Fig.2).

3. For each block centered at pixel (i,j), compute the *x-signature*, $X[0]$, $X[1]$, ... $X[l-1]$, of the ridges and valleys with the oriented window, where

$$X(k) = \frac{1}{w}\sum_{d=0}^{w-1} N(u,v), \quad 0 \le k \le l-1 \qquad (9)$$

and the co-ordinates,

$$u = i + (d - \tfrac{w}{2})\cos\theta'(i,j) + (k + \tfrac{l}{2})\sin\theta'(i,j) \qquad (10)$$

$$v = j + (d - \tfrac{w}{2})\sin\theta'(i,j) - (k - \tfrac{l}{2})\cos\theta'(i,j) \qquad (11)$$

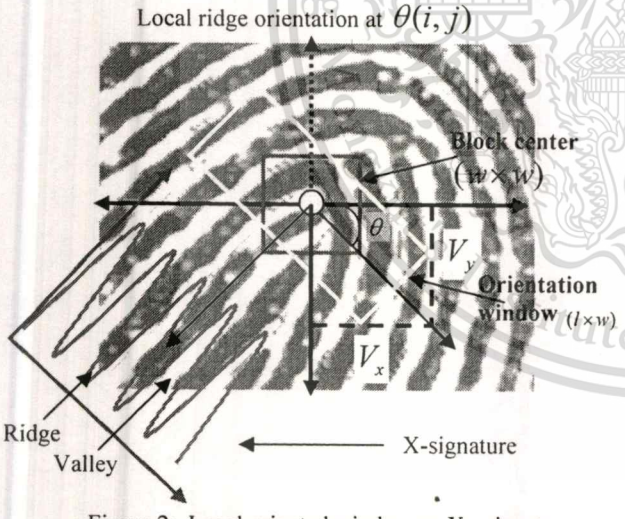


Figure 2: Local oriented window an X - signature

If there is no minutiae and core points in the local oriented window, the x-signature forms a sinusoidal shape wave which has the same frequency as that of the ridges and valleys in the oriented window see Fig. 2for x-signature [4].

4. For the FBI scanning standard of 500 dpi, this ranges is1/3 to 1/25. Therefore, if the estimated value of the frequency is out of this range, then the frequency is assigned a value of -1 to indicate that a valid frequency cannot be obtained.

5. If the estimated inter-ridge pitch is outside this range, or if the *x*-signature does not form a well-defined sinusoidal wave, the estimated frequency of the block is rejected. If there are not too many of these blocks, their local frequencies can be interpolated from the frequencies of the neighboring blocks [7].

2.5 Directional Filtering

Gabor filter is employed to remove noise and preserve the ridge and valley structures. The sinusoidal-shaped waves of ridges and valleys vary slowly in a local constant orientation. Therefore, a band-pass filter that is tuned to the corresponding frequency and orientation can efficiently remove the undesired noise and preserve the true ridge and valley structures. The Gabor filter have both frequency-selective and orientation-selective properties and have optimal joint resolution in both spatial and frequency domains [3, 4, 5].

$$h(x,y:\phi,f) = \exp\left[-\frac{1}{2}\left(\frac{x_\phi^2}{\delta_x^2}+\frac{y_\phi^2}{\delta_y^2}\right)\right]\cos(2\pi f x_\phi) \qquad (12)$$

where

$$x_\phi = x\cos\phi + y\sin\phi \qquad (13)$$

$$y_\phi = -x\sin\phi + y\cos\phi \qquad (14)$$

Where, ϕ is the orientation of the Gabor filter, f is the frequency of a sinusoidal plane wave, δ_x and δ_y are the standard deviations of the Gaussian envelope along x and y axes, respectively.

2.6 Segmentation

The process is to extract the printed image from its background. The consecutive block of 20×20 pixels was designed in this study.

3. Core point detection technique

There exist several techniques for core point detection. For the sage of simplicity we consider the Poincare one.

Poincare Technique (PC)

The Poincare technique can be used to identify both core point and delta point. Upon the availability of estimated orientation field $\theta'(i,j)$ given above, for the pixel in the sub block centered at (i,j) we can compute Poincare index, $PC(i,j)$,

$$PC(i,j) = \frac{1}{2\pi}\sum_{k=0}^{Np}\Delta(k) \qquad (15)$$

$$\Delta(k) = \begin{cases} \delta(k) & if\ |\delta(k)| < \pi/2 \\ \pi + \delta(k) & if\ |\delta(k)| < -\pi/2 \\ 0 & otherwise \end{cases} \qquad (16)$$

and,

$$\delta(k) = \xi(x_{(k+1)_{\bmod Np}}, y_{(k+1)_{\bmod Np}}) - \xi(x_k, y_k) \qquad (17)$$

for particular number of point N_p. The number of points used in the experiment is 8. The core point should yield the Poincare index between 0.45-0.55. If the Poincare index is less than –0.33 then such a block is the delta block and, fifth the center of the block with the value of one is considered a core point.

4. Experiment results

We used the downloaded DB1_A of FVC-2004 as our database in our study. There are 800 fingerprint images captured with optical sensor, "V300" by CrossMatch [6]. An image size is of 640×480 pixels with 500 dpi resolutions. With eyes observation, we found that 746 images hold core point. We fed these 746 images into the Poincare procedure for core point detection. There is no need to perform directional filtering to this set of the images. In contrast, for those that require further enhancement, we did ridge frequency estimation and directional filtering before core point detection. Shown in Fig. 3 a, b, and c, are the resulted core point detection of no-enhancement images whilst d, e, and f, are the corresponding images but with directional filtering.

At a glance, the filtered images can greatly reduce the alias core points. Further investigation to those 746 images, 185 images hold 2 core points (dual core) and 561 images hold single core point (many of them in this group hold left-loop and right-loop patterns). With this observation, we performed the experiment separately. The result of those with 2 cores is shown in Fig.4. And the result of those with single core is shown in Fig. 5. For those with 2 cores (shown in Fig. 4), we classified the obtained results into *i)* 2-core detected correctly (no alias at all), *ii)* 2-core detected but together with some aliases, *iii)* only one core can be detected and some aliases present, *iv)* no core detected but only with aliases, and *v)* no point detect at all (fail case). Similarly we divided the results of those 1 core images (561 images) as shown in Fig. 5. Similar results were obtained.

From the obtained results shown in Fig.4 and Fig.5, we can conclude that the correct detection of the core point could be improved significantly when directional filtering is applied. With such an attempt, alias point detection could reduce. The filter has removed noise effectively. The ridge and valley are greatly enhanced, such that the core point detection method can work with less error. On the negative side, we have to pay for the cost of ridge frequency estimation and directional filtering computation. Based on preliminary investigation by averaging 10 input images, we found that the filtering procedure has taken about 2.5 times greater than non-filtering option, (3.2 GHz Pentium 4 CPU with 512 M RAM; 21.135 seconds and 8.106 seconds respectively).

Figure 3: Core points detection results of various fingerprints image: First row, (a), (b) and (c) demonstrate the core point marking of the image with no application of directional filtering. Second row images; (d), (e) and (f) are the corresponding images shown in the first row, but with the application of directional filtering.

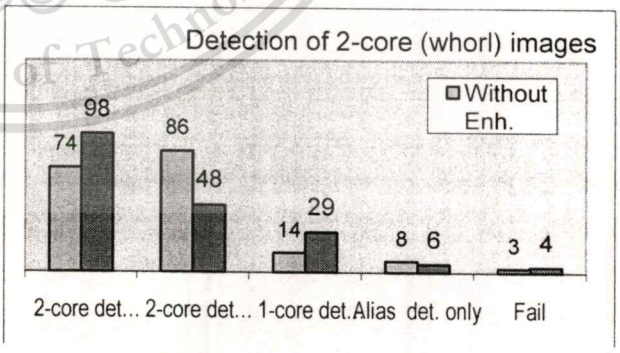


Figure 4: Detection results of 2-core point fingerprint pattern (whorl) experimented without and with directional filtering

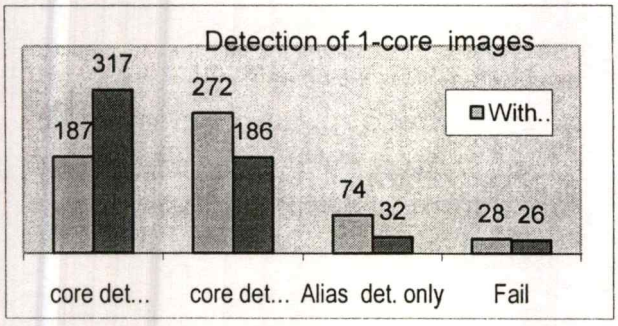


Figure 5: Detection result of 1-core point fingerprint pattern (most are left-loop and right-loop) experimented without and with directional filtering.

Using the directional filter, we could improve the correctness of core point detection. In many cases as shown in Fig.6 below, the algorithm still went wrong. Most failures are because of the really poor quality of images (completely dark and/or completely dry in some image portions). We perhaps need more refinement procedures in considering these particular cases.

Figure 6: Example of image patterns that cause algorithm failed

5. Conclusion

In this paper, we proposed core point identification with local enhancement method. The experiment results clearly show the merit of using directional filtering in the enhancement process. However, we have achieved that with the cost of additional computation. We still need more clever techniques to eliminate aliasing cores as well as techniques for dealing with no information area (completely black or white). The accomplishment would be of benefit in fingerprint classification using geometry structure of core and delta points.

References

[1] Atipat Julasayvake and Somsak Choomchuay, "A Combined Technique in Fingerprint Core Point detection," Proc. of International Workshop on Advanced Image Technology 2007 (IWAIT-2007), Thailand, January 2007, pp. 556-560.

[2] Sen Wang and Yangsheng Wang., "Fingerprint Enhancement in the Singular Point Area," *IEEE signal processing letters*, vol. 11, no. 1, pp. 16-19, January 2004

[3] A.K. Jain S. Prabhakar L. Hong and S. Pankanti, "Filterbank-Based Fingerprint Matching," *IEEE Trans. Image Processing*, vol. 9, no. 5, pp. 846-859, 2000.

[4] L. Hong, Y. Wand, and A. Jain, "Fingerprint image enhancement: Algorithm and performance evaluation," IEEE Pattern Anal. Mach. Intell., vol. 20, no. 8, pp. 777–789, Aug. 1998.

[5] B. G. Sherlock, D. M. Monro, and K. Millard, "Fingerprint enhancement by directional Fourier filtering," *Proc. Inst. Electr. Eng., Vis. Image Signal Process.*, vol. 141, no. 2, pp. 87–94, 1994.

[6] D. Maio, D. Maltoni, R. Cappelli, J. Wayman, and A. Jain, "FVC 2004: Third fingerprint verification competition," in *Proc. Int. Conf. Biometric Authentication*, Hong Kong, Jul. 2004, pp. 1–7.

[7] C. Klimanee and DT Nguyen "On the Design of 2-D Gabor Filtering of Fingerprint Images," Proc. IEE, 2004.

BMEiCON 2009

August 13-14, 2009 Phuket, Thailand

PROGRAM AND ABSTRACTS

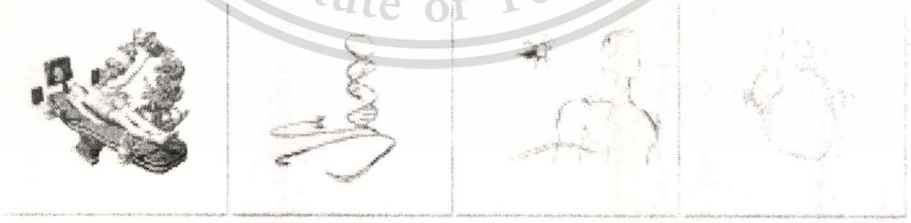
The 2nd Biomedical Engineering International Conference

BMEiCON 2009

August 13-14, 2009 Phuket, Thailand

PROGRAM AND ABSTRACTS

The 2nd Biomedical Engineering International Conference



PERFORMANCE EVALUATION OF FIELD SMOOTHING FILTERS

Keokanlaya Sihalath*, Somsak Choomchuay*, Shatoshi Wada†, and Kazuhiko Hamamoto††

* Department of Electronics, Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang (KMITL), Bangkok, Thailand
Tel: +66-2326-4222, Fax +66-2-739-2398, E-mail: keokanlaya@gmail.com, and kchsomsa@kmitl.ac.th
†School of Information and Communication, Meiji University, 1-9-1 Eifuku Sugimnami, Tokyo 168-8555, Japan, Email: swada@isc.meiji.ac.jp
††Department of Information Media Technology, School of Information Technology and Electronics, Tokai University, Tokyo, Japan, Email: hama@keyeki.cc.u-tokai.ac.jp

ABSTRACT

In this paper, we propose a performance evaluation of orientation field smoothing according to the application of two different filters; average filter and Gaussian filters. Each is investigated at different sizes. The finding is that by applying the filters size of 2-4 times of ridges spacing, the orientation field can be smoothed with less distortion. Gaussian filter seems to offer slightly better performance compared to an average one. Success in core point detection using Poincare technique is used to measure the filtering performance.

Keyword: Fingerprint, type of filter, field smoothing, enhancement, core point detection.

1. INTRODUCTION

Fingerprint is one of the most important biometric technologies. This is because it holds many desirable features such as universality, permanence, collectability, and distinctiveness. Personal identification based on fingerprint matching is now popular in wide range of applications. Most automatic fingerprint identification systems are based in minutiae matching [1, 2, 3]. Frequency domain content also applicable in such a purpose [4].

A fingerprint is the pattern of ridges and valleys on the surface of a fingerprint. Minutiae are local discontinuities in the fingerprint pattern. The most important ones are ridge ending and ridge bifurcation. Spurious ridge structure may change the individuality of input fingerprints. Ridges and valleys in a local neighborhood form a sinusoidal-shaped plane wave, which has a well-defined frequency and orientation. The core point has played important roles in many fingerprints identify techniques. The success of the identification (or matching) process is very much relied on the image quality. In many cases, fingerprints are with numerous discontinuous ridges (dry, wet, damped, scars and smudges). The main difficulty for feature extraction is that fingerprint quality is often too low, thus noise and contrast deficiency can produce false minutiae or hide valid ones. Even high quality images can also yield false minutiae, for example, when the person has cuts or scars in his/her fingers.

There appear many algorithms and techniques proposed and applied to fingerprint image enhancement: using Fourier transform [5], Gabor filters [2], Wavelet transform [6], and minutiae filtering, applied to binary [7] or gray-scale images [8]. The goal of an enhancement algorithm is to improve the clarity of ridge structures of fingerprint images in recoverable regions and to remove the unrecoverable regions.

Core point and delta point are ones of fingerprints' features that commonly used in fingerprint classification [9,10,11,12]. Classification process is of more importance when one wants to search a particular pattern in a large database. There are also many techniques used in searching for the corepoint; for instance, direction of curvature (DC), geometry region (GR), and Poincare method. Among these, poincare technique can also detect a delta point. The Poincare technique accomplishes its task by checking the field orientation around the considering point. The success of point identification is solely depended on the correctness of the field orientation. In many cases, fields are distorted by the discontinuity of the ridges. Smoothing the field is then helpful when one tries to eliminate the alias points created according to filed distortion.

The orientation field can be reliably estimated using 2-D low pass filter, gradient based approach and filter based approaches [4]. The filter based approaches are good as accurate as gradient based because of number of filters window size is so large and the local ridge orientation (field smoothing) varies slowly in a local neighborhood where no core point appears. The discontinuity of ridge and valley due to noise and blur an image can be removing by applying a Gaussian low pass filter. The Average (mean) filter smooth fingerprint image data, thus eliminating noise. This filter performs spatial filtering on each individual pixel in an image using the grey level values in a square or rectangular window surrounding each pixel.

Gabor filter, on the other hands, depends very much on the orientation and ridge frequency. Because the local

orientation changes very rapidly in the core point area, we almost cannot make it accurate. So, the result of enhancement in the core point is fairly poor. There comes the need of prior field smoothing. Hence, there have been many methods attempt to solve this problem. In particular case, Hong et al, [4] proposed a decomposition method to estimate the orientation field from a set of filtered images obtained by applying a bank of Gabor filters on the input fingerprint images.

In this paper, we propose core point detect method where the field smoothing enhancement is needed in prior. In section 2, we review fingerprint enhancement steps as follows: normalization, type of filter, orientation field, ridge frequency and direction filter. In section 3, we review an algorithm used for core point detection, a Poincare technique. In section 4, we compare the obtained results indicating the performance of applying the local enhancement, and finally we concluded the paper in section 5.

2. FINGERPRINT ENHANCEMENT

Many acquired fingerprint images may accompany noises, and also may be are of low quality such as dry, wet, damped, scars, smudges and so on. In most case, the degradation occurs in part. Without any attempts, it is hard to identify the core point and minutiae of such images with those degradations.

2.1 Normalization

Let, *I(i, j)* denote the gray-level value at pixel (*i, j*), *M* and *VAR* denote the estimated mean and variance of image *I*, respectively, and *N(i, j)* denote the normalized gray-level value at pixel (*i, j*). The normalized image is defined as: [12]:

$$N(i,j)=\begin{cases} M_0+\sqrt{\dfrac{VAR_0(I(i,j)-M)^2}{VAR}} & \text{if } I(i,j)>M \\ M_0-\sqrt{\dfrac{VAR_0(I(i,j)-M)^2}{VAR}} & \text{otherwise} \end{cases} \quad (1)$$

Where, M_0 and VAR_0 are the desired mean and variance values, respectively. In this work, we set values mean $M_0=0.5$ and variance $VAR_0=1$. The mean and variant of a gray-level fingerprint image with the dimension of $M\times N$ pixels, are defined respectively as:

2.2 Type of filter.

Orientation field tends to be broken for the disconnected ridge. Spurious may also well observed for the fairly dry image. These are considered as noise. These noises can be removed by applying a low pass filter. At this state we are considering average filter and Gaussian filter to tune up the field orientation.

2.2.1 Average filter

The average filtering is simply to replace each pixel value in an image with the average (mean) value of its neighbors, including itself. This has the effect of eliminating pixel values which are unrepresentative of their surroundings. Average filtering is usually thought of as a convolution filter. Like other convolutions it is based around a kernel, which represents the shape and size of the neighborhood to be sampled when calculating the average. All element of the matrix *i.e.* a_{ij} is 1.

$$G(x,y)=\left(\frac{1}{m}\right)^2\begin{bmatrix} a_{11} & a_2 & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mm} \end{bmatrix} \quad (2)$$

2.2.2 Gaussian filter

The Gaussian filter is used as a low pass filter to blur an image. The Gaussian filter is to use this 2-D distribution as a point-spread function, and this is achieved by convolution. Since the image is stored as a collection of discrete pixels we need to produce a discrete approximation to the Gaussian function before we can perform the convolution. In theory, the Gaussian distribution is non-zero everywhere, which would require an infinitely large convolution kernel, but in practice it is effectively zero more than about three standard deviations from the mean, and so we can truncate the kernel at this point. σ

$$G(x,y)=\left(\frac{1}{2\pi\sigma}\right)^2\exp\left[-\frac{(x^2+y^2)}{2\sigma^2}\right] \quad (3)$$

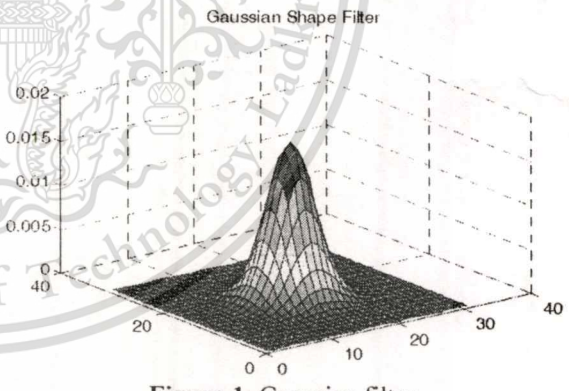


Figure 1. Gaussian filter

2.3 Orientation field

The ridge's orientation is an intrinsic character of the fingerprint image and the pixel, an orientation image, θ, where $\theta'(i,j)$ represents the local ridge orientation at pixel (i, j). Local ridge orientation (field smoothing) is usually specified for a block rather than at every pixel; an image is divided into a set of $w\times w$ non-overlapping blocks and a single local ridge orientation is defined for each block. Note that in a fingerprint image, there is no difference between a local ridge orientation of $\pi/2$ and $3\pi/2$, since the ridges oriented at $\pi/2$ and the ridges

oriented at $3\pi/2$ in a local neighborhood cannot be differentiated from each other.

2.3.1 Field estimation

A number of methods for orientation field estimation have been proposed [2] to estimate the orientation of fingerprint. In this paper, the smoothed orientation field based on least mean square algorithm is summarized as follows:

Divide the input image I into consecutive (non-overlapping) blocks with size $w \times w$

Compute the x and y magnitude of the gradients [($V_x(i,j)$ and $V_x(i,j)$], at each pixel $\partial_x(u,v)$ and $\partial_y(u,v)$ in x and y directions respectively.

Apply the 2D Low Pass Weiner filter, to reduce noise from the fingerprint image, on the x and y gradients.

$$V_x = \sum_{u=i-w/2}^{i+w/2} \sum_{v=j-w/2}^{j+w/2} 2\partial_x(u,v)\partial_y(u,v), \quad (4)$$

$$V_x = \sum_{u=i-w/2}^{i+w/2} \sum_{v=j-w/2}^{j+w/2} 2\partial_x^2(u,v)\partial_y^2(u,v), \quad (5)$$

Estimate the local orientation of each block centered at pixel (i,j) using the following equations.

$$\theta(i,j) = \tfrac{1}{2}\tan^{-1}\left(\frac{V_y(i,j)}{V_x(i,j)}\right) \quad (6)$$

Where, $\theta(i,j)$ is the least square estimate of the local ridge orientation of the block centered at pixel (i,j).

2.3.2 Field smoothing

At this state, the local ridge orientation (field smoothing) varies slowly in a local neighborhood where no core point appears. The discontinuity of ridge and valley due to noise can be softening by applying a low pass filter.

Apply a 2D Gaussian low pass filter the orientation image must be converted to a continuous vector field, which defined as bellows:

$$\Phi_x(i,j) = \cos\{2\theta(i,j)\}, \text{ and } \Phi_y(i,j) = \sin\{2\theta(i,j)\}, \quad (7)$$

where Φ_x and Φ_y are x and y components the vector field, respectively.

With the resulted vector field, the low-pass filters can be performed.

$$\Phi_x' = \sum_{u=-w_\Phi/2}^{w_\Phi/2} \sum_{v=-w_\Phi/2}^{w_\Phi/2} G(u,v)\Phi_x(i-uw-vw), \quad (8)$$

$$\Phi_y' = \sum_{u=-w_\Phi/2}^{w_\Phi/2} \sum_{v=-w_\Phi/2}^{w_\Phi/2} G(u,v)\Phi_y(i-uw-vw), \quad (9)$$

here $G(u,v)$ is a two dimension specified filter of the size $w_\Phi \times w_\Phi$

The smoothed orientation field (local ridge orientation at *(i,j)*) can then be computed as follows:

$$\theta'(i,j) = \frac{1}{2}\tan^{-1}\left(\frac{\Phi_y'(i,j)}{\Phi_x'(i,j)}\right) \quad (10)$$

2.3.3 Ridge frequency

In the gray-level fingerprint image, along ridge and valley can be model as a sinusoidal-shaped wave along the direction perpendicular to the local ridge orientation. Let $N(i,j)$ be the normalized image and $\theta'(i,j)$ be the smoothed orientation field image, and then the steps involved in local ridge frequency estimation are as follows.

Divide $N(i,j)$ image into blocks of size $w \times w$, 5×5 for which center at pixel (i,j).

For each block centered at the pixel (i,j), compute an oriented window of size $l \times w$ (32×5).

For each block centered at pixel (i,j), compute the *x-signature*, $X[0]$, $X[1]$, ... $X[l$ - 1], of the ridges and valleys with the oriented window, where

$$X[k] = \frac{1}{w}\sum_{d=0}^{w-1} N(u,v), \quad 0 \le k \le l-1 \quad (11)$$

and the co-ordinates,

$$u = i + (d - \frac{w}{2})\cos\theta'(i,j) + (k + \frac{l}{2})\sin\theta'(i,j), \quad (12)$$

$$v = j + (d - \frac{w}{2})\sin\theta'(i,j) - (k - \frac{l}{2})\cos\theta'(i,j), \quad (13)$$

If there are no minutiae and core points in the local oriented window, the x-signature forms a sinusoidal-shape wave, which has the same frequency as that of the ridges and valleys in the oriented window x-signature. Let *T(i, j)* be the average number of pixels between two consecutive peaks in the x-signature, then the frequency, $f(i,j)$, is computed as:

$$f(i,j) = 1/T(i,j) \quad (14)$$

If no consecutive peaks can be detected from the x-signature, then the frequency is assigned a value of -1 to differentiate it from the valid frequency values.

For the FBI scanning standard of 500 dpi, this ranges is1/3 to 1/25. Therefore, if the estimated value of the frequency is out of this range, then the frequency is assigned a value of -1 to indicate that a valid frequency cannot be obtained.

If the x-signature does not form a well-defined sinusoidal wave, the estimated frequency of the block is rejected. If there are not too many of these blocks, their

local frequencies can be interpolated from the frequencies of the neighboring blocks.

2.3 Directional Filtering

In normal fingerprint image, the sinusoidal-shaped waves of ridges and valleys vary slowly in a local constant orientation. Therefore, a band-pass filter that is tuned to the corresponding frequency and orientation can efficiently remove the undesired noise while preserving the true ridge and valley structures. The Gabor filter have both frequency-selective and orientation-selective properties and have optimal joint resolution in both spatial and frequency domains.

$$h(x,y:\phi,f)=\exp\left[-\frac{1}{2}\left(\frac{x_\phi^2+y_\phi^2}{\delta_x^2+\delta_y^2}\right)\right]\cos(2\pi f x_\phi) \qquad (15)$$

where

$$x_\phi = x\cos\phi + y\sin\phi \qquad (16)$$

$$y_\phi = -x\sin\phi + y\cos\phi \qquad (17)$$

Where, ϕ is the orientation of the Gabor filter, f is the frequency of a sinusoidal plane wave, δ_x and δ_y are the standard deviations of the Gaussian envelope along x and y axes, respectively.

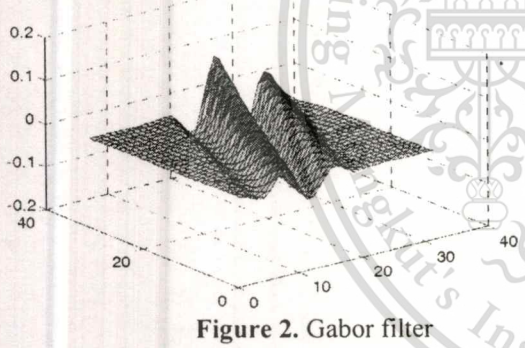


Figure 2. Gabor filter

2.4 Segmentation

The process is to extract the printed image from its background. The consecutive block of 15×15 pixels was designed in this study.

3. CORE POINT DETECTION TECHNIQUES

The core point is generally recognized as the top-most point that the ridges making turns. In order to detect the fingerprint center point area, we first locate the core point corresponding to the uppermost point contained in the inner-most ridge line. There exist several techniques for core point detection. They are; for instance; Poincare index (PC), Direction of curvature (DC), and Geometry region (GR). Each holds its individual complexity and performance.

3.1 Poincare Index Technique (PC)

The Poincare one is fairly simple and suitable for both core point and delta point identify both core point and delta point. Upon the availability of estimated orientation field $\theta'(i,j)$ given above, for the pixel in the sub block centered at (i,j) we can compute Poincare index, $PC(i,j)$, for particular number of point N_p.

$$PC(i,j)=\frac{1}{2\pi}\sum_{k=0}^{Np}\Delta(k) \qquad (18)$$

$$\Delta(k)=\begin{cases}\delta(k) & if\ |\delta(k)|<\pi/2\\ \pi+\delta(k) & if\ |\delta(k)|<-\pi/2\\ 0 & otherwise\end{cases} \qquad (19)$$

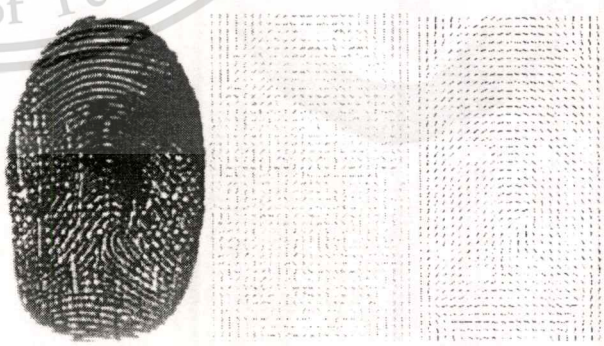
and,

$$\delta(k)=\xi(x_{(k+1)_{\bmod Np}},y_{(k+1)_{\bmod Np}})-\xi(x_k,y_k), \qquad (20)$$

The number of points used in our experiment is 8. The core point should yield the Poincare index 0.5. If the Poincare index is less than –0.5 then such a block is the delta block.

4. EXPERIMENT RESULTS

We used the downloaded DB1_A of FVC-2004 as our database in this study. Captured with optical sensor, "V300" by CrossMatch, an image size is of 640 × 480 pixels with 500 dpi resolutions. In our experiment we randomly chose 104 fingerprint images; all are with core point. In the process of field smoothing, two types of filter, namely, average filter and Gaussian filter were investigated; each at different window size, from 3×3 to 45×45 pixels. Gabor filter is used in the directional filtering process. The performance if the filter is measured by the success in core point locating. Illustrated in Fig. 3, we can see that the field smoothing could have a vital impact on field orientation rectifying.



(a) (b) (c)

Figure 3. Filtering effects (a) original image (b) corresponding orientation field, and (c) orientation field after the application of Gabor filtering

Core point detection results of the fingerprints image with already at high quality is shown in Fig. 4. In Fig. 4 (a) and (b), are the resulted core point detection of no

enhancement images whilst (c) and (d), are the corresponding images but with directional filtering enhancement. It is clear that the enhancement is not so helpful in this case. This is because the images are already at high quality.

Figure 4. Core point detection results of high quality fingerprints image; (a) without enhancement and (b) with enhancement.

In contrast when the images are of low quality, the enhancement by directional filter is essentially useful. This is illustrated in Fig. 5. Shown in (a) and (b), the algorithm has detected many wrong core points when the images are not filtered. However those alias points can be reduced dramatically when the same images are enhanced these are shown in Fig. 5; (c) and (d). In this particular case, Gaussian filter with the widow size of 31 × 31 pixels was applied in the field smoothing process.

Further investigation to those 104 images, 40 images hold 2 core points (dual core) and 64 images hold single core point (many of them in this group hold left-loop and right-loop patterns). To investigate the proper window size of the filter, we counted the number of alias core points while changing the filter size. Results taken upon dual core group and single core group are shown in Fig. 6 and Fig. 7 respectively.

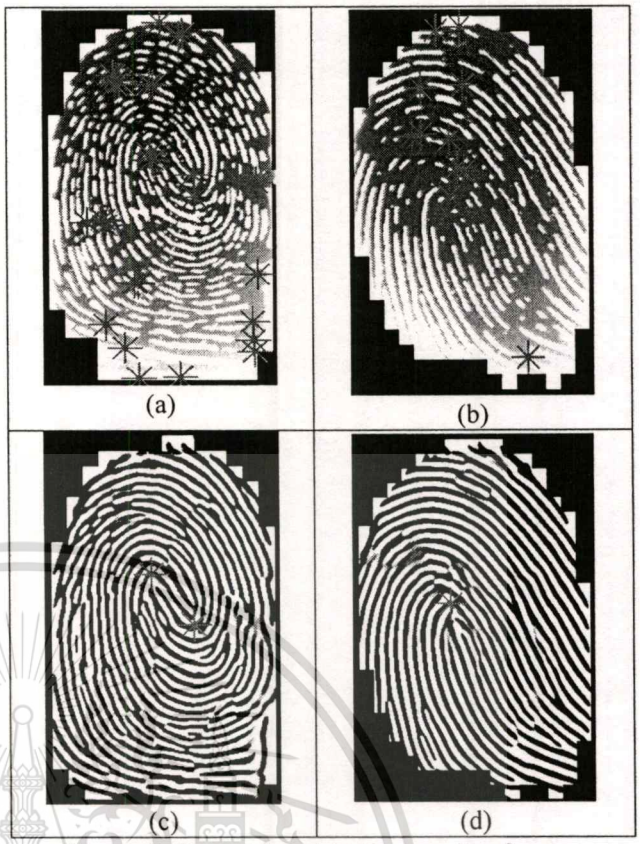


Figure 5. Core points detection results of some fingerprints image: Shown in the first row, (a) and (b) are images with no enhancement. Shown in the second row; (c) and (d) are the corresponding images, but with the application of orientation field smoothing and directional enhancement.

In both cases, Gaussian filter is slightly better than average filter. The filtering effect of both types did not shown much improvement when the window size is over 31× 31 pixels. However, we have noticed that when the filter window size is too large (more than 45 × 45 pixels), the filtered image is deteriorated as the detected core point seems to drift apart (for the dual core images).

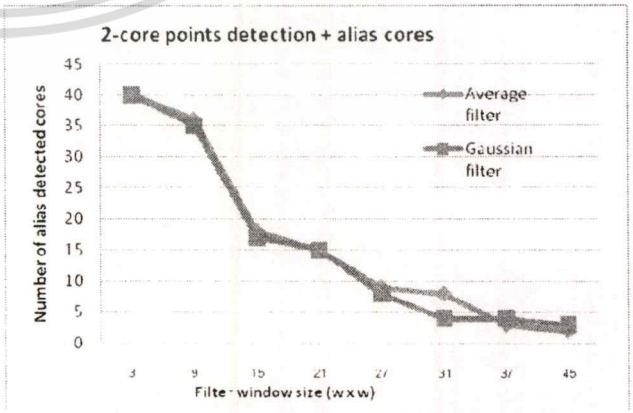


Figure 6. Alias core points versus filter widow size. Images are 2-core point fingerprint pattern (Whorl). Total is 40 images.

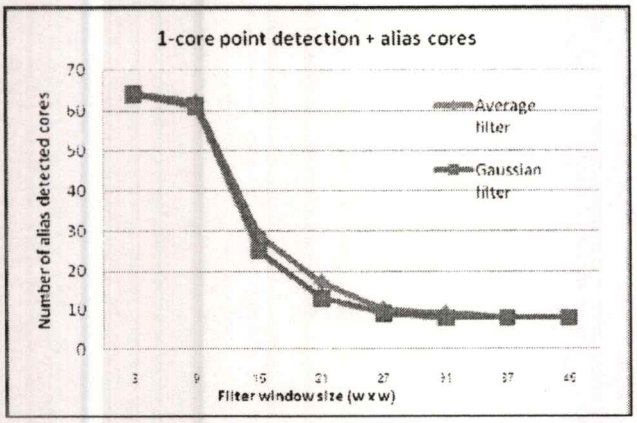


Figure 7. Alias core points versus filter widow size. Images are single-core point fingerprint pattern (left-loop or right-loop). Total is 64 images.

One may also notice that we cannot get the number of alias core point down to zero, and even worst we also got a number of wrong located core points in our experiment. This is true. The main reason is that some images are of so poor quality. They are beyond the performance of the enhancement process. The cases are shown in Fig. 8. We still need more procedure to eliminate these alias points.

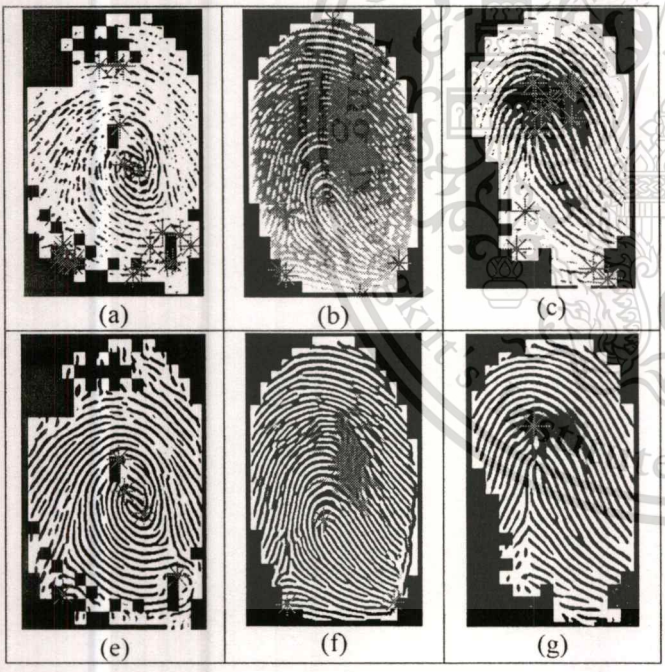


Figure 8. Example of very poor quality images, shown in the first row, (a), (b) and (c) are images with no enhancement and the second row are those with the application of enhancement.

5. CONCLUSION

Average filtering and Gaussian filtering for field smoothing are investigated in this work. A Gaussian filter is slightly superior compared to an average filter at all window size. Filtering effect becomes stable when the window size is greater than 31× 31 pixels. The smaller widow size cannot rectify well the noisy field. As a result many alias core points still left. The too large window size is also not desirable as the filter ridge seems to separate apart. The right core point detection together with directional filtering are useful in fingerprint matching in the frequency domain. Both core point and delta point detection are useful in fingerprint classification.

ACKNOWLEDGEMENT

This work is partially supported by SEED-Net/AUN Program. Authors would like to express our gratitude and sincere thanks to SEED-Net project.

REFERENCES

[1] Jain, L. Hong, and R. Bolle, "On-line fingerprint verification," IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(4):302–313, April 1997.

[2] L. Hong, Y. Wan, and A. Jain. "Fingerprint image enhancement: Algorithm and performance evaluation," IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(8):777–789, August 1998.

[3] Y. He, J. Tang, X. Luo, and T. Zhang, "Image enhancement and minutiae matching in fingerprint verification," Pattern Recognition Letters, 24, 2003.

[4] A.K. Jain S. Prabhakar L. Hong and S. Pankanti, "Filterbank-Based Fingerprint Matching," IEEE Trans. Image Processing, vol. 9, no. 5, pp. 846-859, 2000.

[5] B. G. Sherlock, D. M. Monro, and K. Millard. "Fingerprint enhancement by directional Fourier Filtering," IEEE Proceedings in Visual Image Signal Processing, 141(2):87–94, April 1994.

[6] M. Tico, P. Kuosmanen, and J. Saarinen "Wavelet Domain Features for Fingerprint Regcognition," Electronic Letter, Vol.37, No.1, Jan. 2001.

[7] A. Farina, Z.M. Kovacs-Vajna, and A. Leone, "Fingerprint minutiae extraction from skeletonized binary images," Pattern Recognition, 32(5):877–889, 1999.

[8] D. Maio, D. Maltoni, "Direct Gray-Scale Minutiae Detection in Fingerprints," IEEE Trans. Pattern Anal. Machine Intell., Vol. 19, No. 1, pp. 27-40, 1997.

[9] A. K. Jain, S. Prabhakar and L. Hong, "A Multichannel Approach to Fingerprint Classification," IEEE Trans. On PAMI, Vol.21, No.4, pp. 348-359, April 1999.

[10] Sen Wang, Wei Wei Zhang ,Yang Sheng Wang, "Fingerprint Classification by Directional Fields", Proc. 4th IEEE Int. Conf. on Multimodal Interfaces (ICMI'02), 2002.

[11] A. Julasayvake and S. Choomchuay, "A Combined Technique in Fingerprint Core Point detection," Proc. Of Int. Workshop on Advanced Image Technology 2007, Thailand, January 2007, pp. 556-560

[12] A. Julasayvake and S. Choomchuay, "An Algorithm for Fingerprint Core Point detection," Proc. of Int. Symp. On Signal Processing and its Applications 2007 (ISSPA-2007), United Arab Emirates, February 2007.

---

1 While submitting this work, Satoshi Wada is on a sabbatical leave to the department of Electronics, Faculty of Engineering, KMTIL.

ISSN: 2086-4868

Directional Filtered Fingerprint Images:

An Investigation and Evaluation of Minutiae Consistency

Keokanlaya Sihalath [*1], Somsak Choomchuay [*2], and Kazuhiko Hamamoto [**]
[*] Department of Electronic Engeering, Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang
Chalongkrung Road, Ladkrabang, Bangkok 10520, Thailand
[*1] e-mail: keokanlaya @gmail.com [*2] email: skcxch@gmail.com;
[**]Information Media Technology, School of Information Technology and Electronics
Tokai University, Tokyo, Japan
e-mail: hama@keyeki.cc.u-tokai.ac

Abstract—**In this paper, we investigate the consistency of minutiae that could be effected by the application of directional filtering during the fingerprint image enhancement process. Two types of similar filters: Gabor filter and Second derivative of Gaussian filters are employed for investigation. These filters are also used in the scheme of directional wavelet transform and pyramid technique. The commonly-well-known database FVC-2004 is used in this study. Based on the obtained results we can conclude that ridge end point and bifurcation is influenced by the filtering application. Moreover, the condition of the input image is also very important for the designed filter to work efficiently.**

Keywords- Fingerprint enhancement; Minutiae; Directional filtering; Biometric

I. INTRODUCTION

Among the uses of biometric in personal identification, fingerprint-base identification has been use for a very long time. A fingerprint is the pattern of ridges and valleys on the surface of a finger. Minutiae are local discontinuities in the fingerprint pattern. The most important ones are ridge ending and ridge bifurcation. Spurious ridge structure may change the individuality of input fingerprints. The core point as well as the delta point(s) that are always known as singular points have played important roles in many fingerprints identifies techniques [1, 2, 3]. Not less, minutiae-based finger print matching is also widely used [4, 5]. The correct detection of endpoint (ridge-ending), junction (ridge-bifurcation), core point and delta point is therefore very important. This is very much relied on the image quality. In many cases, fingerprint with numerous discontinuous ridges (dry, wet, damped, scars and smudges) can cause errors in fingerprint identification process. Noise and contrast deficiency can produce false minutiae or hide valid ones. Even high quality images can also yield false minutiae, for example, when the person has cuts or scars in his/her fingers.

The main objective of an enhancement process is to improve the clarity of ridge structures of fingerprint images in recoverable regions and to remove the unrecoverable regions. Fingerprint enhancement can be applied to either a gray-level fingerprint image [6] or a binary one [7]. As ridges and valleys in a fingerprint image alternate and run parallel to each other in a local neighborhood. Ridges and valleys in a local neighborhood form a sinusoidal-shaped plane wave, which has a well-defined frequency and orientation. These specific features of a fingerprint image have lead to several image filtering and enhancement techniques.

Of with the ridge direction and ridge frequency of the finger print image, there appear many algorithms and techniques proposed for image enhancement. These are, for instance, using Fourier Transforms [8, 9], Gabor filters [4, 10], Wavelet transform [11, 12, 13]. On the positive side, the filter tries to remove noise, enhance the ridge contrast, and try to repair the broken ridge. On the negative side, the filter wrongly joined the too-close ridge. This can produce the wrong bifurcation. If those broken ridges cannot be joined, the unnecessary line-ends are detected. Minutiae-base matching is then fail or the matching score is lower.

In this paper, we investigate the effect of directional filter applied in fingerprint image enhancement. We would like to know whether the minutiae (such as ridge end and bifurcate could be enhanced or reversely be destroyed. The rest of this paper is organized as follows: In section II, we outline the enhancement process that includes; normalization, directional field estimation and smoothing, directional filtering, and thinning for minutiae detection. In section III, directional filters are elaborated in quite detail. Directional wavelet transform and pyramid technique are also included. Experiment and results are given in section IV. In this section, ridge end and bifurcation are observed separately. Finally we concluded the paper in section V.

II. ENHANCEMENT PROCESS

A. Normalization

This step is quite common in many image processing routines. The effort can reduce the variance in gray-level values along ridges and valleys by means of adjust the gray-level values to the predefined constant mean and variance. And normalization can remove the influences of sensor noise and gray-level deformation.

This work is partially supported by AUN/SEED-Net program under the Research Collaboration scheme.

B. Frequency Estimation & Field Estimation

We can approximate ridge frequency of the fingerprint image by dividing it into blocks (for instance, pixels). For any local neighborhood which does not have singularities (core or delta), the gray-levels of the pixels form a sinusoidal shape along the direction orthogonal to the ridge orientation. An oriented window (oriented in the direction orthogonal to the local ridge orientation) is used to approximate this sinusoid. The inverse of the average distance between the numbers of peaks encountered is the local frequency of that block. In our case, the ridge frequency of 0.10-0.12 was measured. The obtained frequency is used in the directional filtering step.

Since directional filtering is a direction-sensitive filter. We have to estimate the local orientation before applying the filter. Before such an attempt we apply 2D Low Pass Weiner filter (block of 5×5) to the image as a purpose of noise reduction with some effect of directional filtering.

Let $\phi(i,j)$ be the *orientation field* that represents the local ridge at pixel (i,j). The local ridge is generally specified for a block rather than for every pixel. Thus, an image is divided in to a set of non-overlapping blocks, size of $w\times w$. Each bock holds a single orientation. A procedure proposed for orientation estimation is summarized below [4].

Divide the input image into consecutive (non-overlapping) blocks size of $w\times w$. We used $w=15$.

Compute the x and y magnitude of the gradients $\partial_x(u,v)$ and $\partial_y(u,v)$, at each pixel (i,j).

The local orientation of each block centered at pixel (i,j) can be estimated by:

$$\phi(i,j)=\tfrac{1}{2}\tan^{-1}\left(\frac{V_y(i,j)}{V_x(i,j)}\right) \quad (1)$$

Where, $\phi(i,j)$ is the least square estimate of the local ridge orientation of the block centered at pixel (i,j) and

$$V_x(i,j)=\sum_{u=i-w/2}^{i+w/2}\sum_{v=j-w/2}^{j+w/2}2\partial_x(u,v)\partial_y(u,v), \quad (2)$$

$$V_y(i,j)=\sum_{u=i-w/2}^{i+w/2}\sum_{v=j-w/2}^{j+w/2}2\partial_x^2(u,v)\partial_y^2(u,v), \quad (3)$$

Given in eqn. (1) is just the estimation of orientation field. For some fine applications and an application that involve no directional filter, one can perform field smoothing. The operation requires conversing the current estimated field into continuous vector field that followed by a low pass filtering. In this paper, however, field smoothing is omitted.

C. Directional Filtering (DF)

Based on the estimated field, the directional filter is applied. The gray scale image is enhanced eventually. Two similar filters will be given in more details in the next section. The said filters are extended to corporate with wavelet and pyramid technique.

D. Thinning and Minutiae Allocation

For the enhanced image to be investigated for its minutiae location consistency, the ridges are thinned such that the pattern line width is only a single pixel. Ridge end and bifurcation are then easily located. In contrast, this step in no need for core and/or delta point detection. A well recognized technique such as Poincare is used instead.

III. DIRECTIONAL FILTERING

Directional filtering is of interest when one wants to highlight the image pattern or texture holding particular direction or orientation. Gabor filter is a linear filter used in image processing. Its impulse response is defined by a harmonic function multiplied by a Gaussian function. Similarly, second derivative of Gaussian function modified by multiplying with a cosine function can hold the feature of direction filtering.

A. Gabor Filter

The Gabor filter have both frequency-selective and orientation-selective properties and have optimal joint resolution in both spatial and frequency domains.

$$h(x,y:\phi,f)=\exp\left\{-\tfrac{1}{2}\left(\frac{x_\phi^2+y_\phi^2}{\sigma_x^2+\sigma_y^2}\right)\right\}\cos(2\pi f x_\phi) \quad (4)$$

where

$$x_\phi=x\cos\phi+y\sin\phi \quad (5)$$

$$y_\phi=-x\sin\phi+y\cos\phi \quad (6)$$

Where, ϕ is the orientation of the Gabor filter, f is the frequency of a sinusoidal plane wave, σ_x and σ_y are the standard deviations of the Gaussian envelope along x and y axes, respectively.

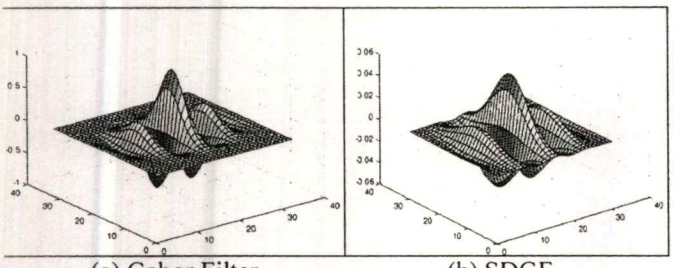
B. Second Derivative Gaussian Filter (SDGF)

This second derivative can enhance the ridge and suppress the valley for certain degree. To enhance the effectiveness of the filter we can modify equation (15) slightly by co-operating the cosine function (or plan wave) as follow [14]:

$$sdg(x,y:\phi,f)=\frac{(x_\phi^2-\sigma_\phi^2)(y_\phi^2-\sigma_\phi^2)}{2\pi\sigma_\phi^{10}}\exp\left(-\frac{x_\phi^2+y_\phi^2}{\sigma_\phi^2}\right)\cos(2\pi f x_\phi) \quad (7)$$

Where, ϕ is the orientation of the second derivative filter, f is the frequency of a sinusoidal plane wave, σ_ϕ is the standard deviations of the Gaussian envelope. x_ϕ and y_ϕ define the x and y axes of the filter coordinate frame.

Appearance of Gabor filter and second derivative filter is shown in Fig. 1 below. They are actually in the same class [15].



(a) Gabor Filter (b) SDGF

Figure 1. Appearance of filters

(Gabor filter: $\sigma_x = 0.4$, $\sigma_y = 0.5, \phi = 0^{\circ}, f = 0.12$)

(SDGF: $\delta_\phi = 0.4$, $\phi = 0^{\circ}, f = 0.12$)

C. Directional Wavelet Transform

The discrete wavelet transform of function $f(x,y)$ of size $M \times N$ is formulated as:

$$W_\varphi(j_0,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\varphi_{j_0,m,n}(x,y) \quad (8)$$

$$W_\psi^i(j,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\psi_{j,m,n}^i(x,y) \quad (9)$$

where $i = \{H,V,D\}$, $\psi^i(x,y)$ are directional wavelet, j_0 is the starting scale of a scaling function $\varphi(x,y)$, and the $W_\varphi(j_0,m,n)$ coefficients define the approximation of $f(x,y)$, at scale j_0 . The $W_\psi(j,m,n)$ coefficients represent the horizontal, vertical and diagonal details for scales $j \geq j_0$. Here $j_0 = 0$ and select $M + N = 2^J$ so that $j = 0,1,2,..,J-1$ and $m, n = 0,1,2,..,(2^J - 1)$. Then $f(x,y)$ is obtained via the inverse discrete wavelet transform.

$$f(x,y) = \frac{1}{\sqrt{MN}} \sum_m \sum_n W_\varphi(j_0,m,n)\varphi_{j_0,m,n}(x,y) + \frac{1}{\sqrt{MN}} \sum_{i=H,V,D} \sum_{j=j_0}^{\infty} \sum_m \sum_n W_\psi^i(j,m,n)\psi_{j,m,n}^i(x,y), \quad (10)$$

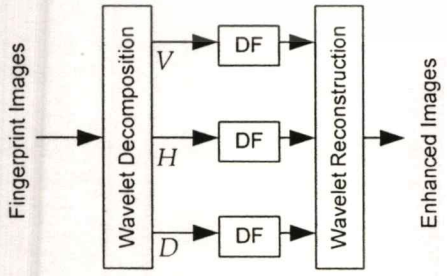


Figure 2. Fingerprint enhancement by mean of wavelet transform and directional filtering

In this work, a directional wavelet transform is applied to decompose the image into its orientation representation [16]. Directional filtering is applied to each direction before image reconstruction. This is shown in Fig. 2 below. We have applied the Gabor filter and the second derivative of Gaussian filter in the DF state.

D. Pyramid Technique

As a multi-resolution processing, the image pyramid has been applied for fingerprint enhancement lately [15, 17]. The Gaussian pyramid decomposes the image into several bands. It acts as a low pass filter. In contrast, the Laplacian pyramid is actually equivalent to band pass filtering in the spatial domain.

Shown in Fig. 2 below, in the decomposition phase the image is down sampled by half at each level. 3-4 levels are generally sufficient for most applications. At each level, a directional filter is applied and the image $L_i(m_i,n_i)$ is obtained. There is no need to apply the filter to the final state (last level).

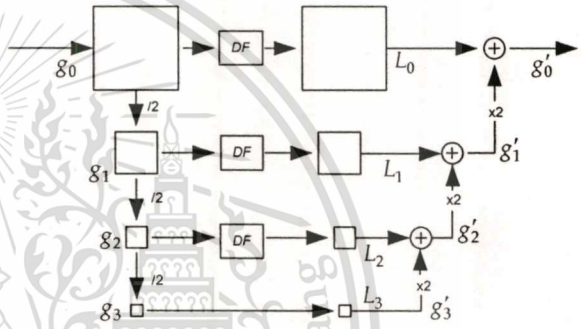


Figure 3. Pyramid technique in fingerprint image enhancement

In the construction phase, the stored image is up sampled by the factor of 2 and added to the lower level image of which the same dimension. The said operation is applied successively until the original dimension is reached and the enhanced image is obtained.

IV. Experiment & Results

We used the downloaded DB1_A of FVC-2004 as our database in this study [18]. Captured with optical sensor, "V300" by CrossMatch, an image size is of 640×480 pixels with 500 dpi resolutions. In our experiment we randomly chose 6 fingerprint images; all are quite varied in image quality. However, all holds core point. Orientation field estimation is applied after image normalization. Two type of directional filters; namely Gabor filter and Second derivative of Gaussian filter are investigated together with other two arrangements which are directional wavelet transform technique and pyramid technique. The filtered image is then thinned for the ease of ridge end and bifurcation observation. The obtained results are given in Table I and Table II. Table I demonstrates the ridge end points whilst table II demonstrates the bifurcation. Shown in Fig. 4 are the example images of various process steps.

ISSN: 2086-4868

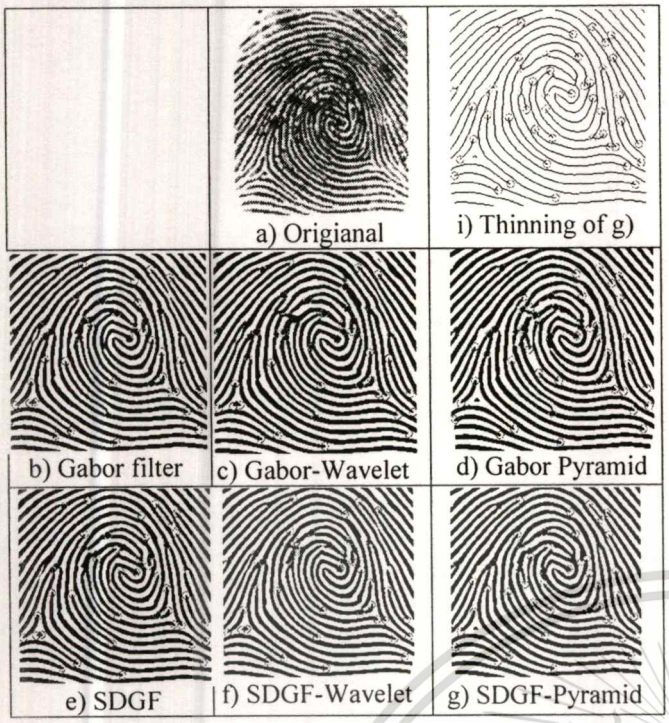


Figure 4. Images at various process step

Shown in Table I and II, our counting is based on the genuine point. The positive or negative value means how big the difference the number of points it holds compare with the number of genuine points. It is quite clear that corporation of the filter with either wavelet transform or pyramid technique can offer better performance as the number of ridge end point decreases when compared to that of applying the filter only. However, one also can notice that the filter application can destroy the genuine end point. Such a situation is demonstrated by the negative difference. This can happen because the too-close ridges are joined. According to Table I, Image #2 is greatly influenced by the said conditions.

Application of the filter for image enhancement seems to have less effect on bifurcation. In many cases, the destroyed line-ends are because of their joining and a bifurcation is generated. If one looks into Table II, the said conditions are revealed. Image #2 (see also Fig. 6) holds some thicker line near its end point the space between the considering line and the neighboring line is then lower. According to the field smoothing, those line are joined and a bifurcation is generated.

TABLE I. RIDGE END DETECTION

	Images					
	#1	#2	#3	#4	#5	#6
Genuine Ridge End	13	9	5	13	23	8
Gabor Filter	+1	0	-1	+1	+1	0
Gabor-Wavelet	0	-1	0	-1	+1	-1
Gabor-Pyramid	-1	-1	+1	+1	0	+1
SDGF	+1	-1	+1	+1	0	0
SDGF-Wavelet	-1	-2	+1	+1	+1	0
SDGF-Pyramid	-1	-1	0	+1	+1	+1

TABLE II. BIFURCATION DETECTION

		Images					
		#1	#2	#3	#4	#5	#6
	Genuine Bifurcate	8	13	13	16	13	16
After Enhancement	Gabor Filter	0	-1	0	-1	+1	0
	Gabor-Wavelet	+1	0	0	0	-1	0
	Gabor-Pyramid	+1	0	-1	0	-1	0
	DSGF	-1	0	0	0	0	0
	SDGF-Wavelet	0	+2	0	-1	0	0
	SDGF-Pyramid	0	0	0	-1	-1	0

SDGF=Second Derivative of Gaussian Filter

Shown in Fig. 5, we have circled some obvious incomplete filtering that result in wrong ridge end decision. In the upper and left circles, an island (should contain 2 ridge ends) is marked as 1 ridge end and 1 bifurcate. For the middle circle, the situation is more complicate. Perhaps without eye-observe made to the actual pattern it is fairly hard to identify the ride end and bifurcation.

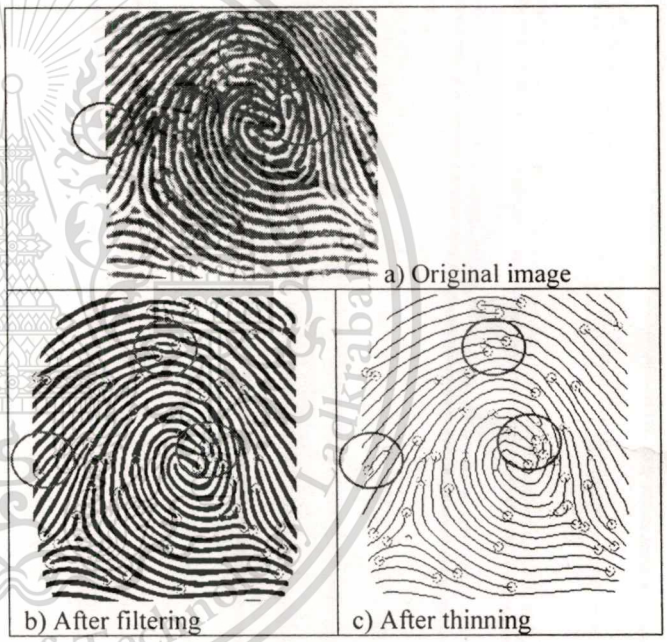


Figure 5. Some imperfect filterings than can cause wrong decision

CONCLUSIONS

In this paper we have studied the consistency of minutiae of fingerprint images enhanced by the use of directional filtering. Two similar filters: Gabor filter and second derivative of Gaussian filter are investigated. The corporation of these two filters in Directional wavelet arrangement and pyramid technique enhancement are also involved. Despite the computation complexity, the extra arrangement of the filter can perform the broken line connection. However, another obvious disadvantage is that: it does wrong join the too-close ridge. As a result, it sometimes produces some fake bifurcations. For a poor quality image and even an expert do the examination, it is

really difficult to justify for the interest feature. In many cases, the broken lines still cannot be joined by the filter. The fake ridge ends still exist. The window size used during the field estimation process is also important in such a way that; on one side too big window can give the good field smoothing but it can ignore the bifurcation. On the other side the too small window size cannot join well the broken ridge but it pick up well those bifurcations. We conclude our observation that designing of the filter is fairly critical. Its performance can be good for some type of poor images but may not so efficient for others. In most cases directional filter can remove noise dramatically.

ACKNOWLEDGMENT

This work is partially supported by AUN/SEED-Net Program. Authors would like to express our gratitude and sincere thanks to SEED-Net project.

REFERENCES

[1] A.K. Jain S. Prabhakar L. Hong and S. Pankanti, "Filterbank-Based Fingerprint Matching," *IEEE Trans. Image Processing*, vol. 9, no. 5, pp. 846-859, 2000.

[2] A. Julasayvake and S. Choomchuay, "An Algorithm for Fingerprint Core Point detection." *Proc. of Int. Symp. on Signal Processing and its Applications 2007 (ISSPA-2007)*, United Arab Emirates, February 2007.

[3] Sen Wang and Yangsheng Wang., "Fingerprint Enhancement in the Singular Point Area," *IEEE signal processing letters*, vol. 11, no. 1, pp. 16-19, January 2004.

[4] L. Hong, Y. Wan, and A. Jain. "Fingerprint image enhancement: Algorithm and performance evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(8):777–789, August 1998.*

[5] Y. He, J. Tang, X. Luo, and T. Zhang, "Image enhancement and minutiae matching in fingerprint verification," *Pattern Recognition Letters*, 24, 2003.

[6] D. Maio, D. Maltoni, "Direct Gray-Scale Minutiae Detection in Fingerprints," *IEEE Trans. Pattern Anal. Machine Intell., Vol. 19, No. 1, pp. 27-40, 1997.*

[7] A. Farina, Z.M. Kovacs-Vajna, and A. Leone, "Fingerprint minutiae extraction from skeletonized binary images," Pattern Recognition, 32(5):877–889, 1999.

[8] B. G. Sherlock, D. M. Monro, and K. Millard. "Fingerprint enhancement by directional Fourier Filtering," *IEEE Proceedings in Visual Image Signal Processing, 141(2):87– 94, April 1994.*

[9] S. Chikkerur and V. Govindaraju, "Fingerprint image enhancement using STFT analysis," in *Proc. Int. Workshop on Pattern Recognition for Crime Prevention, Security and Surveillance*, 2005, pp. 20–29.

[10] J. Yang, L. Liu, T. Jiang, and Y. Fan, "A modified gabor filter design method for fingerprint image enhancement". *Pattern Recognition Letters*, 24:1805–1817, 2003.

[11] Zhengmao Ye, Habib Mohamadian, and Yongmao Ye, "Information Measures for Biometric Identification via 2D Discrete Wavelet Transform," *Proceedings of the 3rd Annual IEEE Conference on Automation Science and Engineering Scottsdale, AZ, USA, Sept 22-25, 2007.*

[12] Anto Melvin Paul and R. Mary Lourde,"A Study on Image Enhancement Techniques for Fingerprint Identification" . *Proceedings of the IEEE International Conference on Video and Signal Based Surveillance , AVSS'06.*

[13] Wei-Peng, Zhang, Qing-Ren. Wang, Y. Tang, "A wavelet-based method for fingerprint image enhancement". *Proceeding of the First International Conference on Machine Learning and Cybernetics, Beijing, 4-5 November 2002*

[14] F. Yue, W.Zuo, K.Wang, and D. Zhang, "A Performance Evaluation of Filter Design and Coding Schemes for Palmprint Recognition," *Proc. of ICPR 2008.*

[15] S. Choomchuay and K. Sihalath, "An Application of Second Derivative of Gaussian Filters in Fingerprint Image Enhancement," *Proc. of iCBBE-2010.*

[16] K. Sihalath, S. Choomchuay, S. Wada, and K. Hamamoto, "Fingerprint Image Enhancement with Second Derivative Gaussian Filter and Directional Wavelet Transform," *Proc. of ICTTA-2010.*

[17] H. Fronthaler, K. Kollreider, and J. Bigun, "Local Features for Enhancement and Minutiae Extraction in Fingerprints," *IEEE Trans. on Imag. Proc.* Vol.17, No.3, March 2008, pp. 354-363.

Finger print Verification Contest 2002; FVC2002: Available at http://bias.csr.unibo.it/fvc2002/

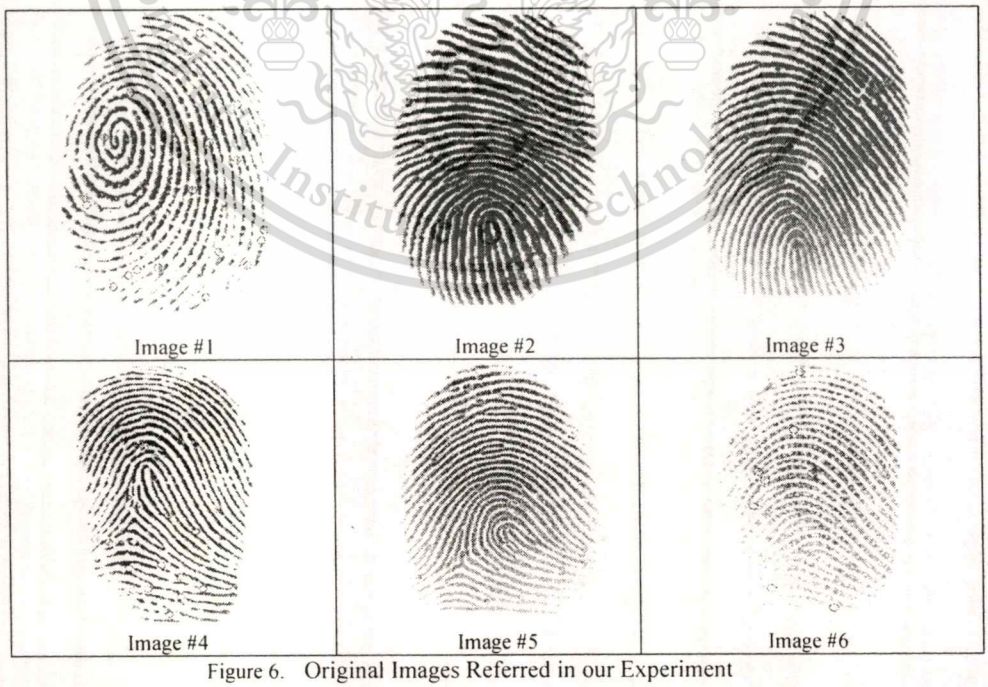
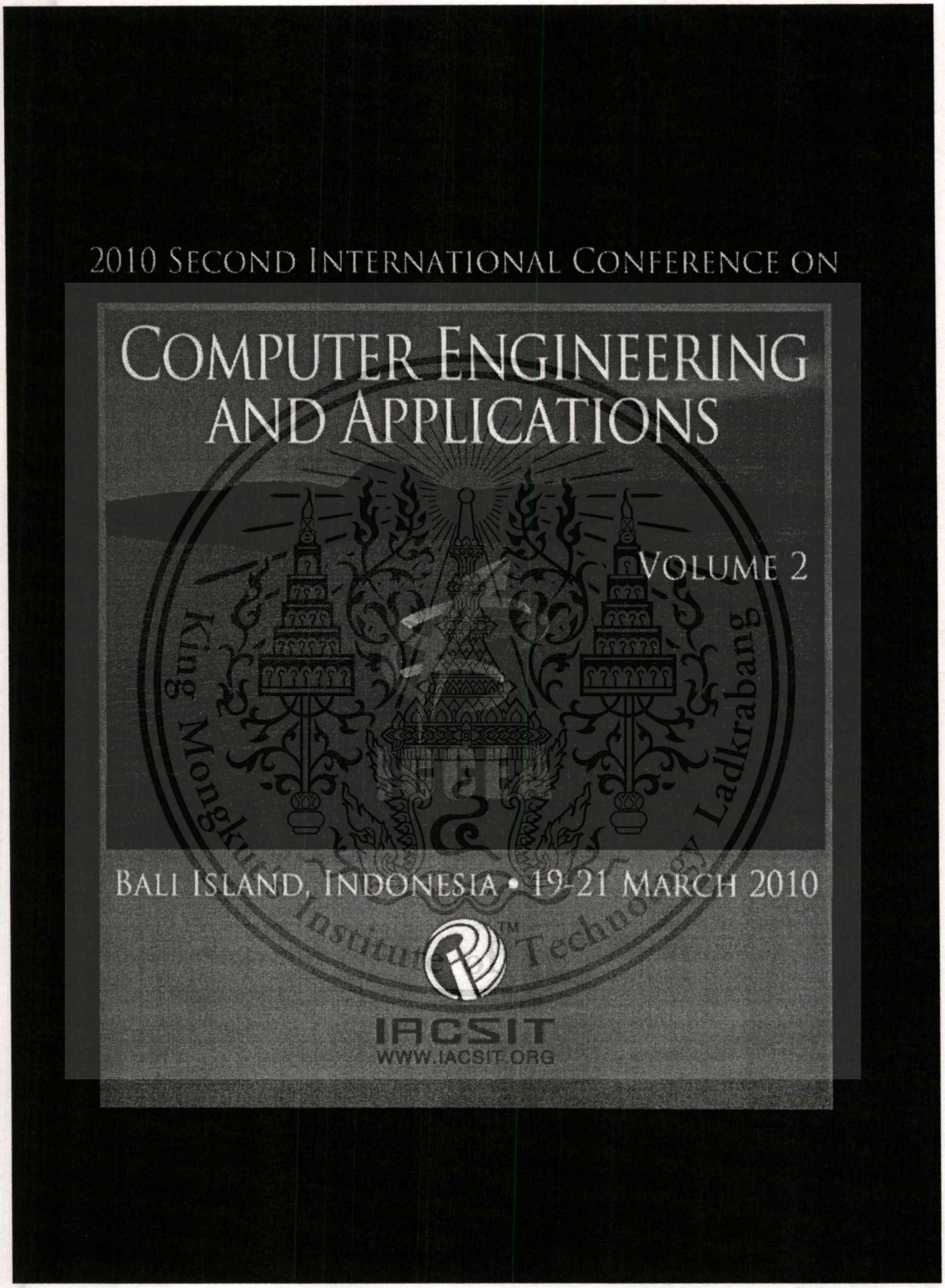


Figure 6. Original Images Referred in our Experiment



Fingerprint Image Enhancement with Second Derivative Gaussian Filter and Directional Wavelet Transform

Keokanlaya Sihalath†, Somsak Choomchuay†
† Department of Electronics, Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang (KMITL), Bangkok, Thailand
E-mail: keokanlaya@gmail.com, kchsomsa@kmitl.ac.th

Shatoshi Wada††, and Kazuhiko Hamamoto†††
†† School of Information and Communication, Meiji University, Tokyo, Japan
††† Information Media Technology, School of Information Technology and Electronics, Tokai University, Tokyo, Japan
Email: swada@isc.meiji.ac.jp, hama@keyeki.cc.u-tokai.ac

***Abstract*— In this paper, we propose a technique for enhancing the quality of fingerprint images. Directional wavelet transform and second derivative of a Gaussian filter are applied. The original fingerprint image is decomposed into approximation and detail sub-images. To each sub-dimension a directional filter: second derivative of Gaussian filter is applied for tuning up the image features. The enhanced image is measured for its improvement by testing the success of core point identification where Poincare technique is used. The commonly-well-known database FVC-2004 is used in this study. The obtained results offer clean visualization as well as the increase the success of true core point detection.**

Keywords-Fingerprint enhancement; Wavelet transform; the second derivative of Gaussian filter; Biometric.

I. INTRODUCTION

Among all the biometrics, fingerprint-based identification is one of the most popular and reliable biometric techniques. This is because it holds many desirable features such as universality, permanence, collectability, and distinctiveness. Personal identification based on fingerprint matching is now popular in wide range of applications. Most automatic fingerprint identification systems are based in minutiae matching [1, 2, 3].

A fingerprint is the pattern of ridges and valleys on the surface of a fingerprint. Minutiae are local discontinuities in the fingerprint pattern. The most important ones are ridge ending and ridge bifurcation. Spurious ridge structure may change the individuality of input fingerprints. Ridges and valleys in a local neighborhood form a sinusoidal-shaped plane wave, which has a well-defined frequency and orientation. The core point has played important roles in many fingerprints identify techniques. The success of the identification (or matching) process is very much relied on the image quality. In many cases, fingerprints are with numerous discontinuous ridges (dry, wet, damped, scars and smudges). The main difficulty for feature extraction is that fingerprint quality is often too low, thus noise and contrast deficiency can produce false minutiae or hide valid ones. Even high quality images can also yield false minutiae, for example, when the person has cuts or scars in his/her fingers.

There appear many algorithms and techniques proposed and applied to fingerprint image enhancement: using Fourier transform [4, 5], Gabor filters [2, 6], Wavelet transform [7, 8, 9, 10,11], and minutiae filtering, applied to binary [12] or gray-scale images [13]. The main of an enhancement algorithm is to improve the clarity of ridge structures of fingerprint images in recoverable regions and to remove the unrecoverable regions.

In the fingerprint image, the pattern is related to the ridge direction. In principle, the enhancement can help visualizing the ridges. It is reasonable to use any type of directional filter in either single or multi-resolution analysis. In this work, a directional wavelet transform is applied to decompose the image into its orientation representation. Directional filtering is applied to each direction before image reconstruction. This is shown in Fig. 1 below. Unlike many filters reported in the literature, we have applied the second derivative of Gaussian filter in the DL state.

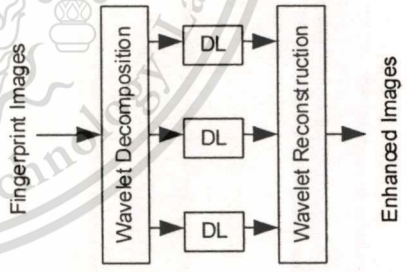


Figure 1: Fingerprint enhancement by mean of wavelet transform and directional filtering

The second derivative of Gaussian filter is the straightforward extension of the Gaussian (and also first derivative) filter. It can be applied independently in each dimension. The second derivative filter also holds the features that it depends very much on the orientation and ridge frequency. By nature, the local orientation changes quite rapidly in the core point area (sometimes also delta point), we almost cannot make it accurate. So, the result of enhancement in the ridge in such an area is fairly poor. There have been many methods attempt to solve this problem [14].

978-0-7695-3982-9/10 $26.00 © 2010 IEEE
DOI 10.1109/ICCEA.2010.178

The rest of this paper is organized as follows. In section II, the directional wavelet transform is again given in brief. Several steps in enhancement procedure in cluing transform and filtering are given in section III. In section IV, to evaluate the quality of enhanced image, we count the success in core point detection. A Poincare technique is reviewed. In section V, we demonstrate the obtained results indicating the performance of applying the directional wavelet transform second derivative filtering, and finally we concluded the paper in section VI.

II. DIRECTIONAL WAVELET TRANSFORM

Wavelet transform is suited for the analysis of transient and time varying signals. In two dimensions, a scaling function $\varphi(x,y)$, and three directional wavelets $\psi^H(x,y)$, $\psi^V(x,y)$ and $\psi^D(x,y)$ are necessary. Each scaling function or wavelet is the product of the one dimensional scaling function φ and corresponding wavelet ψ. The four two-dimensional products produce the scaling function (1) and separable directional sensitive wavelets (2), (3) and (4).

$$\varphi(x,y)=\varphi(x)\varphi(y) \quad (1); \qquad \psi^H(x,y)=\psi(x)\varphi(y) \quad (2)$$

$$\psi^V(x,y)=\varphi(x)\psi(y) \quad (3); \qquad \psi^D(x,y)=\psi(x)\psi(y) \quad (4)$$

These wavelets measure the gray level variations for images along three directions, where $\psi^H(x,y)$ measures variations along columns (horizontal). $\psi^V(x,y)$ responds to variations along rows (vertical) and $\psi^D(x,y)$ corresponds to variations along diagonals.

In a two dimensional discrete wavelet transform, the scaled and translated basis functions are defined by:

$$\varphi_{j,m,n}(x,y)=2^{j/2}\varphi(2^j x-m,2^j y-n), \quad (5)$$

$$\psi^i_{j,m,n}(x,y)=2^{j/2}\psi^i(2^j x-m,2^j y-n), \quad i=\{H,V,D\}, \quad (6)$$

where index i identifies the directional wavelets according to equation (2), (3) and (4). The discrete wavelet transform of function $f(x,y)$ of size $M\times N$ is formulated as:

$$W_\varphi(j_0,m,n)=\frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\varphi_{j_0,m,n}(x,y) \quad (7)$$

$$W^i_\psi(j,m,n)=\frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\psi^i_{j,m,n}(x,y) \quad ;i=\{H,V,D\} \quad (8)$$

where $i=\{H,V,D\}$, j_0 is the starting scale, the $W_\varphi(j_0,m,n)$ coefficients define the approximation of $f(x,y)$, at scale j_0. The $W_\psi(j,m,n)$ coefficients represent the horizontal, vertical and diagonal details for scales $j\ge j_0$. Here $j_0=0$ and select $M+N=2^J$ so that $j=0,1,2,..,J-1$ and $m,\ n=0,1,2,..,(2^J-1)$. Then $f(x,y)$ is obtained via the inverse discrete wavelet transform.

$$f(x,y)=\frac{1}{\sqrt{MN}}\sum_m\sum_n W_\varphi(j_0,m,n)\varphi_{j_0,m,n}(x,y)$$

$$+\frac{1}{\sqrt{MN}}\sum_{i=H,V,D}\sum_{j=j_0}^{x}\sum_m\sum_n W^i_\psi(j,m,n)\psi^i_{j,m,n}(x,y), \quad (9)$$

Now the wavelets are defined by both the scaling function $\varphi(x,y)$ (Father Wavelet) and wavelet functions $\Psi(x,y)$ (mother wavelet) in the discrete time domain. The wavelet function acts as a bandpass filter whose bandwidth is reduced to half after each scaling. At the level 1, rows of $f(x,y)$ are lowpass and highpass filtered and downsampled. Then columns of the row filtered images are lowpass and highpass filtered and downsampled similarly. The two dimensional discrete wavelet transform filters the scaling approximation coefficients to construct the scale approximation and detail coefficients. The output of each level always includes: approximation, horizontal detail, vertical detail and diagonal detail. As a result, each of them is a quarter of the size of its original image. Four quarter size output sub-images $(W_\varphi, W^H_\psi, W^V_\psi, W^D_\psi)$ are the inner products of $f(x,y)$ and the 2D scaling and wavelet functions, followed by downsampling by a factor of two [7].

III. WAVELET-BASED FINGERPRINT ENHANCEMENT

One of the most widely cited fingerprint enhancement using wavelet transform and Gabor filtering. This method uses wavelet transform for demising and increases the contrast between the ridge and background (valley) by using a map function to the wavelet coefficient set, and thereafter, the Gabor filter method can further enhance the ridge using the orientation and frequency information [7, 8, 9, 10,11]. In contrast, we alternatively used the modified second derivative Gaussian filter at this state.

In our method, the Daubechies wavelet (db4) is used to decompose the fingerprint image before directional filtering. The second derivative of Gaussian filter is applied directly to each sub-image. We reconstruct the fingerprint image by using the enhanced approximation image and detail images produced in decomposition. The detail process of the enhancement are as follows.

A. Normalization

The processing of fingerprint normalization can reduce the variance in gray-level values along ridges and valleys by means of adjust the gray-level values to the predefined constant mean and variance. And normalization can remove the influences of sensor noise and gray-level deformation.

B. Ridge Frequency

In a gray scale image, repeated ridges and valley appearance of fingerprint patterns can be viewed as a sinusoidal shape with some particular frequency. We can approximate ridge frequency of the fingerprint image by dividing it into blocks (for instance, 5×5 pixels). For any

local neighborhood which does not have singularities (core or delta), the gray-levels of the pixels form a sinusoidal shape along the direction orthogonal to the ridge orientation. An oriented window (oriented in the direction orthogonal to the local ridge orientation) is used to approximate this sinusoid. The inverse of the average distance between the numbers of peaks encountered is the local frequency of that block. In our case, the ridge frequency of 0.10-0.12 was measured. The obtained frequency is used in the directional filtering step.

C. Wavelet Decomposition

Different base function convolution with the image can have different effect in the resolution. In this paper, we select Daubechies to implement the decomposition for its sufficient information in sub-image approximation. Theoretically, we can decompose the image into sub-images at any level. However, too low resolution is not suitable because an excessive down sampling of the signal can vanish the orientation characteristic of the ridge structure. Generally, only one or two decomposition levels are selected [9, 10]. We used only one decomposition level in this experiment. The obtained decomposed images are shown below. The upper right sub-image shows horizontal component. The lower left sub-image shows vertical component while the lower right sub-image shows the diagonal component respectively.

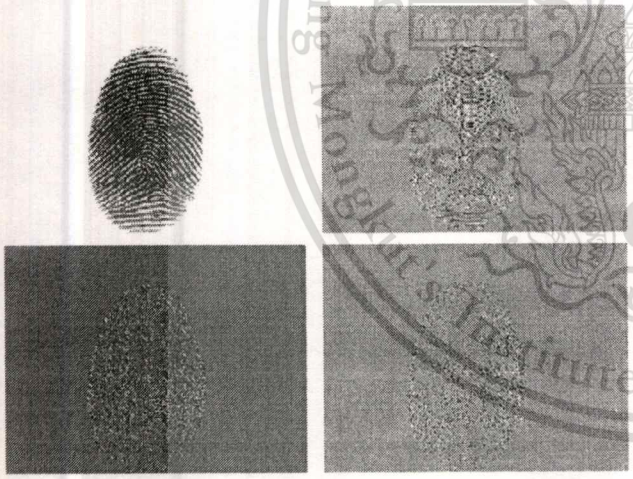


Figure 2: Wavelet transforms decomposition one level.

D. Orientation Field Estimation

Since directional filtering is a direction-sensitive filter. We have to estimate the local orientation before applying the filter. Before such an attempt we apply 2D Low Pass Weiner filter (block of 5×5) to the image as a purpose of noise reduction with some effect of directional filtering.

Let $\phi(i,j)$ be defined as the *orientation field* of a finger print image. $\phi(i,j)$ represents the local ridge at pixel (i,j). In general local ridge is specified for a block rather than for every pixel. Thus, an image is divided in to a set of non-overlapping blocks, size of $w\times w$. Each bock holds a single orientation. A procedure proposed for orientation estimation is summarized below [2].

Divide the input image into consecutive (non-overlapping) blocks size of $w\times w$. We used $w=16$.

Compute the x and y magnitude of the gradients $\partial_x(u,v)$ and $\partial_y(u,v)$, at each pixel (i,j).

The local orientation of each block centered at pixel (i,j) can be estimated by:

$$\phi(i,j)=\tfrac{1}{2}\tan^{-1}\left(\frac{V_y(i,j)}{V_x(i,j)}\right) \tag{11}$$

Where, $\phi(i,j)$ is the least square estimate of the local ridge orientation of the block centered at pixel (i,j) and

$$V_x(i,j)=\sum_{u=i-w/2}^{i+w/2}\sum_{v=j-w/2}^{j+w/2}2\partial_x(u,v)\partial_y(u,v), \tag{12}$$

$$V_y(i,j)=\sum_{u=i-w/2}^{i+w/2}\sum_{v=j-w/2}^{j+w/2}2\partial_x^2(u,v)\partial_y^2(u,v), \tag{13}$$

E. The Second Derivative of Gaussian Filter

In normal fingerprint image, the sinusoidal-shaped waves of ridges and valleys vary slowly in a local constant orientation. Therefore, a band-pass filter that is tuned to the corresponding frequency and orientation can efficiently remove the undesired noise while preserving the true ridge and valley structures [14].

The 2-D Gaussian filter is mostly adopted as an image preprocessing step for image smoothing and denoising. The second derivative of Gaussian filter given in equation (15) is the straightforward extension of the Gaussian first derivative filter. This can be applied independently to each dimension.

$$sdg(x,y:\phi)=\frac{(x_\phi^2-\sigma_\phi^2)(y_\phi^2-\sigma_\phi^2)}{2\pi\sigma_\phi^{10}}\exp\left(-\frac{x_\phi^2+y_\phi^2}{\sigma_\phi^2}\right) \tag{15}$$

$$x_\phi=x\cos\phi+y\sin\phi \tag{16}$$

$$y_\phi=-x\sin\phi+y\cos\phi \tag{17}$$

This second derivative can enhance the ridge and suppress the valley for certain degree. To enhance the effectiveness of the filter we can modify equation (15) slightly by co-operating the cosine function (or plan wave) as follow:

$$sdg(x,y:\phi,f)=\frac{(x_\phi^2-\sigma_\phi^2)(y_\phi^2-\sigma_\phi^2)}{2\pi\sigma_\phi^{10}}\exp\left(-\frac{x_\phi^2+y_\phi^2}{\sigma_\phi^2}\right)\cos(2\pi f x_\phi) \tag{18}$$

Where, ϕ is the orientation of the second derivative filter, f is the frequency of a sinusoidal plane wave, σ_ϕ ($\sigma_\phi=4$) is the standard deviations of the Gaussian envelope. x_ϕ and y_ϕ define the x and y axes of the filter coordinate frame.

F. Wavelet Reconstruction

After modifying the approximation sub-image with the second derivative of Gaussian filter, we can reconstruct the final fingerprint image.

IV. CORE POINT DETECTION TECHNIQUES

The performance of the enhancement method detailed in the previous section is evaluated by an accuracy of core point detection. Among several core point detection method, the Poincare one is fairly simple and suitable for both core point and delta point identifying. Upon the availability of estimated orientation field $\phi(i,j)$ given in the previous section, for the pixel in the sub block centered at (i,j) we can compute Poincare index, $PC(i,j)$, for particular number of point N_p [15, 16].

$$PC(i,j) = \frac{1}{2\pi}\sum_{k=0}^{Np}\Delta(k) \tag{19}$$

$$\Delta(k) = \begin{cases} \delta(k) & if\ |\delta(k)| < \pi/2 \\ \pi + \delta(k) & if\ |\delta(k)| < -\pi/2 \\ 0 & otherwise \end{cases} \tag{20}$$

$$\delta(k) = \xi(x_{(k+1)_{\mathrm{mod}\,Np}}, y_{(k+1)_{\mathrm{mod}\,Np}}) - \xi(x_k, y_k), \tag{21}$$

The number of points used in the experiment is 8. The core point should yield the Poincare index 0.5. If the Poincare index is less than –0.5 then such a block is the delta block, and the center of the block with the value of one is considered as a core point.

V. EXPERIMENT RESULTS

We used the downloaded DB1_A of FVC-2004 as database in our study. There are 800 fingerprint images captured with optical sensor, "V300" by CrossMatch. An image size is of 640×480 pixels with 500 dpi resolutions. With eyes observation, we found that 746 images hold core point. The rest do not. We fed these 746 images into the Poincare procedure for core point detection. Images with wrong core point detection are selected and fed them to the enhancement process where directional wavelet transform and second derivative of Gaussian filter are imposed before core point detection.

In our results of experiment are shown in shown in Fig. 3(a) the fingerprint enhance images seem to look better that the non-enhanced. However, with or without can yield the right core point detection. This is because the original images are fairly of good quality, and shown in Fig. 3(b) first row are original fingerprint images is very poor quality. It can be seen clearly that there are broken ridge gaps in some regions. Second row concept of our method corresponding to directional wavelet transforms. We observe that our proposed second row is improving the quality of fingerprint image obviously. Our proposed algorithm offers better results in enhancing the weak edges of ridge with suppressing the noise and healing the interrupted ridges.

The directional wavelet transform that involve the second derivative of Gaussian filter enhancement can be result in alias core points reducing as shown in Fig. 3(b). Further investigation to those 746 images, 185 images hold 2 core points (dual core) and 561 images hold single core point (many of them in this group hold left-loop and right-loop patterns). With this observation, we performed the experiment separately. The result of those with 2 cores is shown in Fig. 4(a). Moreover, the result of those with single core is shown in Fig. 4(b). We classified the obtained results into: *i)* 2-core detected correctly (no alias at all); *ii)* 2-core detected but together with some aliases, *iii)* only one core can be detected and some aliases present; *iv)* no core detected but only with aliases; *v)* no point detect at all (fail case).

From the obtained results shown in Fig. 4(a) and Fig. 4(b), we can conclude that the correct detection of the core point could be improved when directional wavelet transform that involves the second derivative of Gaussian filter enhancement is applied. With such an attempt, alias point detection could be dramatically reduced. The filter has removed noise, the ridge and valley patterns are enhanced greatly.

VI. CONCLUSION

In this paper, we propose fingerprint enhancement with directional wavelet transform that involves the second derivative of a Gaussian filter. The experiment results indicate that noise in the image could be reduced significantly. As a result, the ridge (and valley) and core point detection of fingerprint could be improved. With less computational complexity the wavelet transform can be omitted and applying only the second derivative of a Gaussian filtering. The images could also be improved but not as good as that involves the directional wavelet transform.

ACKNOWLEDGMENT

This work is partially supported by AUN/SEED-Net Program. Authors would like to express our gratitude and sincere thanks to SEED-Net project.

REFERENCES

[1] A. Jain, L. Hong, and R. Bolle, "On-line fingerprint verification," *IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(4):302–313, April 1997.*

[2] L. Hong, Y. Wan, and A. Jain. "Fingerprint image enhancement: Algorithm and performance evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(8):777–789, August 1998.*

[3] Y. He, J. Tang, X. Luo, and T. Zhang, "Image enhancement and minutiae matching in fingerprint verification," *Pattern Recognition Letters, 24, 2003.*

[4] B. G. Sherlock, D. M. Monro, and K. Millard. "Fingerprint enhancement by directional Fourier Filtering," *IEEE Proceedings in Visual Image Signal Processing, 141(2):87– 94. April 1994.*

S. Chikkerur and V. Govindaraju, "Fingerprint image enhancement using STFT analysis," in *Proc. Int. Workshop on Pattern Recognition for Crime Prevention, Security and Surveillance*, 2005, pp. 20–29.

J. Yang, L. Liu, T. Jiang, and Y. Fan, "A modified gabor filter design method for fingerprint image enhancement". *Pattern Recognition Letters*, 24:1805–1817, 2003.

C.T. Hsieh, E. Lai, and Y.C. Wang, "An effective algorithm for fingerprint image enhancement base on wavelet tranform," *Pattern Recognition* 36 (2003), 303-312, available at: www.elsevier.com.

Zhengmao Ye, Habib Mohamadian, and Yongmao Ye, "Information Measures for Biometric Identification via 2D Discrete Wavelet Transform," *Proceedings of the 3rd Annual IEEE Conference on Automation Science and Engineering Scottsdale, AZ, USA, Sept 22-25, 2007.*

Anto Melvin Paul and R. Mary Lourde,"A Study on Image Enhancement Techniques for Fingerprint Identification" . *Proceedings of the IEEE International Conference on Video and Signal Based Surveillance , AVSS'06.*

0] Wei-Peng, Zhang, Qing-Ren. Wang, YYTang, "A wavelet-based method for fingerprint image enhancement". *Proceeding of the First International Conference on Machine Learning and Cybernetics, Beijing, 4-5 November 2002*

[11] Miao-li WEN, Yan LIANG, Quan PAN, Hong-cai ZHANG, "A Gabor filter based fingerprint enhancement algorithm in wavelet domain". *Communications and Information Technology, 2005. ISCIT 2005. IEEE International Symposium on* 12-14 Oct. 2005. Volume 2, On page(s): 1468- 1471.

[12] A. Farina, Z.M. Kovacs-Vajna, and A. Leone, "Fingerprint minutiae extraction from skeletonized binary images," *Pattern Recognition, 32(5):877–889, 1999.*

[13] D. Maio, D. Maltoni, "Direct Gray-Scale Minutiae Detection in Fingerprints," *IEEE Trans. Pattern Anal. Machine Intell., Vol. 19, No. 1, pp. 27-40, 1997.*

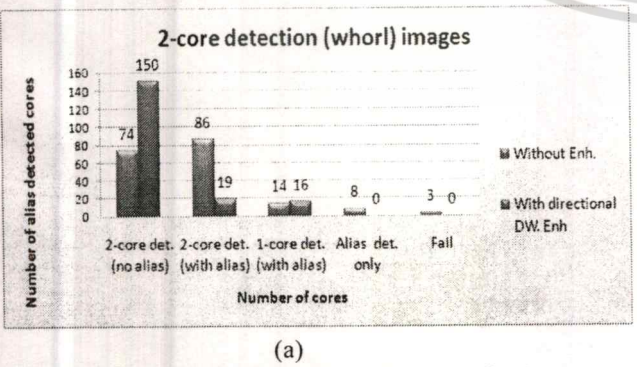
[14] Feng Yue, Wangmeng Zuo, Kuanquan Wang "A Performance Evaluation of Filter Design and Coding Schemes for Palmprint Recognition", *19th International Conference on Pattern Recognition, ICPR 20008, 8-11 Dec, 2008.*

[15] A. K. Jain, S. Prabhakar and L. Hong, "A Multichannel Approach to Fingerprint Classification," IEEE Trans. On PAMI, Vol.21, No.4, pp. 348-359, April 1999.

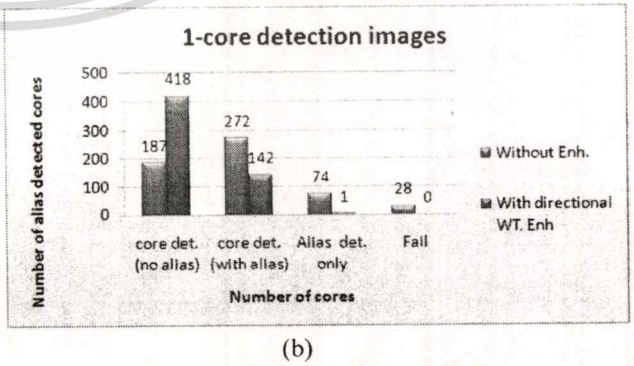
[16] A. Julasayvake and S. Choomchuay, "An Algorithm for Fingerprint Core Point detection," *Proc. of Int. Symp. On Signal Processing and its Applications 2007 (ISSPA-2007), United Arab Emirates, February 2007.*

(a) (b)

Figure 3: Core point location results of some fingerprint images: first row of (a) and (b) are original images, second row are images after enhancement. Set (a) are images with fairly high quality whilst set (b) are poor quality images.



(a)
(185 images hold double core points)



(b)
(561 images hold single core points)

Figure 4: (a) Result of 2-core point fingerprint pattern images and (b) result of 1-core point fingerprint pattern images