CASCADED FILTERING FOR BIOMETRIC IDENTIFICATION USING RANDOM PROJECTION

Thesis submitted in partial fulfillment
of the requirements for the degree of

MS by Research
in
Computer Science

by

Atif Iqbal
200602003
atif.iqbal@research.iiit.ac.in

CENTER FOR VISUAL INFORMATION TECHNLOGY
International Institute of Information Technology
Hyderabad - 500 032, INDIA
June 2012

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "Cascaded Filtering for Biometric Identification using Random Projections" by Atif Iqbal, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date	Adviser: Prof. Anoop Namboodiri

To My Parents

Acknowledgments

I would like to thank my advisor Dr. Anoop Namboodiri for his support and guidance during the past three years. I was fortunate enough to have him as my guide. I sincerely appreciate all his help, generosity, patience and deep insights over perspective solutions to the problems. The freedom of thought and work style provided by him has helped me achieve a sense of responsibility and being in control of my work. His ability to get quality work done easily under pressure of deadline is truly inspirational. I have been able to look up to him for guidance and open discussions on academic matters outside the scope of my thesis. I am also grateful to fellow lab mates at the CVIT, IIIT Hyderabad for their stimulating company during the past three and a half years. I would also like to thank Dr. C.V. Jawahar, Dr. P J Narayan and Dr. Jayanthi Siwaswamy for various references and guidance in different subjects related to the stream.

I would like to thank my CVIT labmates and seniors Pratyush Bhatt, Chhaya Methani, Gopal Dutt Joshi and Maneesh Upmanyu for their support and guidance and support during these years. I would like to thanks my friends Kumar Srijan, Siddharth Choudhary and Vaibhav Kedia for their support during my thesis and course work. I would like thank my friends, always insisted me to work hard and always believe in me. Above all I am thankful to my family members for their patience, support and love.

Abstract

Biometrics has a long history, and is possibly as old as the human race itself. It is often used as an advanced security measure to safeguard important artifacts, buildings, information, etc. Biometrics is increasingly being used for secure authentication of individuals and is making its presence felt in our lives. With the fast increase in computational power, such systems can now be deployed on a very large scale. However, efficiency in large scale biometric matching is still a concern as the problem of deduplication (removal of duplicates) of a biometric database with N entries is $O(N^2)$, which can be extremely challenging for large databases. To make this problem tractable, many indexing methods have been proposed that would speed up the comparison process. However, the expectation of accuracy in such systems combined with the nature of biometric data makes the problem a very challenging one.

Biometric identification often involves explicit comparison of a probe template against each template stored in a database. An effective approach to speed up the process is that of filtering, where a light-weight comparison is used to reduce the database to smaller set of candidates for explicit comparison. However, most existing filtering schemes use specific features that are hand-crafted for the biometric trait at each stage of the filtering. In this work, we show that a cascade of simple linear projections on random lines can achieve significant levels of filtering. Each stage of filtering consists of projecting the probe onto a specific line and removal of database samples outside a window around the probe. The approach provides a way of automatic generation of filters and avoids the need of developing specific features for different biometric traits. The method also provides us with a variety of parameters such as the projection lines, the number and order of projections, and the window sizes to customize the filtering process to a specific application. The experiments are performed on the fingerprints, palmprints and iris.

For both iris and palmprint datasets, the representation that we use (before projection) is the popularly used thresholded filter response from pre-defined regions of the image. Experimental results show that using an ensemble of projections reduce the search space by 60% without increasing the false negative identification rate in palmprint. However for stronger biometrics such as iris, the approach does not yield similar results. We further explore this problem to find a solution, specifically for the case of fingerprints.

The fundamental approach here is to explore the effectiveness of weak features in a cascade for filtering fingerprint databases. We start with a set of potential indexing features computed from minutiae triplets and minutiae quadruplets. We show that by using a set of random lines and the proposed fitness function, one can achieve better results that optimized projection methods such as PCA or LDA. Ex-

perimental results on fingerprint datasets show that using an ensemble of projections we can reduce the penetration to 26% at a hit rate of 99%. As each stage of the cascade is extremely fast, and filtering is progressive along the cascade, one can terminate the cascade at any point to achieve the desired performance. One can also combine this method with other indexing methods to improve the overall accuracy and speed. We present detailed experimental results on various aspects of the process on the FVC 2002 dataset.

The proposed approach is scalable to large datasets due to the use of random linear projections and direcly lends to pipelined processing. The method also allows the use of multiple existing features without affecting the computation time.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

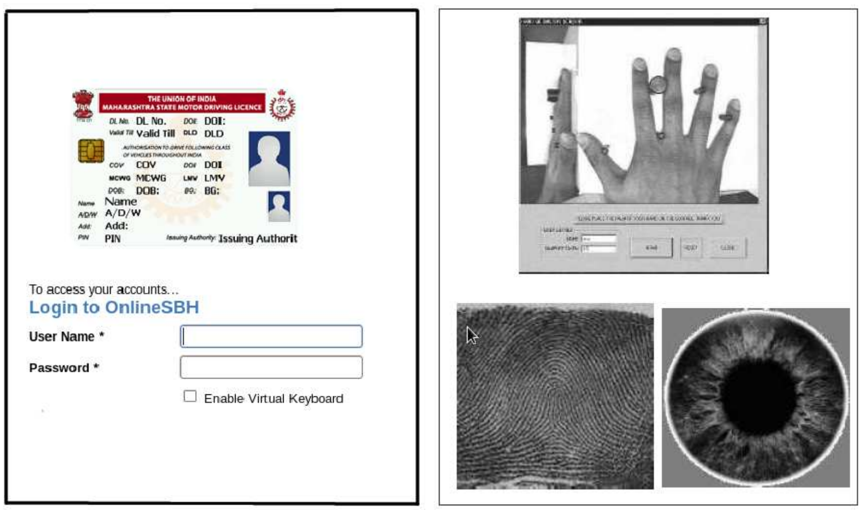
Bometrics is branch of science that aims at uniquely recognising human being based on physiological or behavioral traits. Physiological characteristics are related to the physical appearance of the body. Some examples include fingerprint, face recognition, DNA, palm print, hand geometry, iris recognition. Behavioral are related to the characteristics that we are habituated to perfom such as typing rhythm gait, walking and voice Physiological characteristics includes fingerprint, face, hand/finger geometry, iris, retina, signature, gait, palmprint, voice pattern, ear, hand vein, odor or the DNA information of an individual to establish identity ([11], [75])(See Fig. 1.2). In the biometric literature, these characteristics are referred to as traits, indicators or identifiers. While biometric systems have their own limitations ([60]) they have an edge over traditional security methods in that they cannot be easily stolen or shared. Besides boosting security, biometric systems also enhance user convenience and trust by removing the need to design and remember passwords. Face recognition is the oldest and most basic of a characteristics that is used for human recognition. Human face have been used for recognition since the beginning of the civilisation. Excavations have revealed a 31,000 years old cave painting where the painter has left his hand prints as a signature. Face Recognition is supoosed to be the first form of biometrics. Chinese have started using fingerprint as biometrics in 14^{th} century. Due to rapid growth of city population in the beginning of 18^{th} century, there was a formally recognized need to identify people. The formal use of fingerprints was started in South Africa, Asia and Europe in 18^{th} century by the police department. By the the end of 19^th century indexing method for fingerprint was developed in India by Edward Henry who was working in the Bengal police who started Henry Classification System, was a precursor to the clasification system which was used for many years by the US Federal Bureau of Investigation. A detail timeline of use of biometrics can be found in [69].

The importance of biometrics in modern society has been reinforced by the need for large-scale identity management systems whose functionality relies on the accurate determination of an individuals identity in the context of several different applications. Examples of these applications include sharing networked computer resources, performing remote financial transactions or crossing a border. The proliferation of web-based services (e.g., online banking) and the deployment of decentralized customer service centers (e.g., credit cards) have further underscored the need for reliable identity management systems that can accommodate a large number of individuals. Automatic Biometric system have become available over the last few decades, due to significant improvements in the field of computer processing. Many of these new automated techniques, however are based on the ideas that were originally concieved hundreds, or even thousands of years ago. Although fingerprints are still in use today, the idea to develop automatic system for use our voices, our hands, palms, iris and faces were proposed. In 1936, the idea that our iris patterns are unique was proposed. The development of an iris identification system began in 1993 [37]; in 1994 the first iris recognition algorithm was patented, and the year after that, a commercial product measuring iris became available. In 1960s face recognition became semi-automated [8]. Funding for the project was provided by the unnamed agency and little of their work was published. In 1991, face detection is pioneered making real time face recognition possible [73] by Turk and Pentland. However the algorithm works well with clear background and aligned faces. Face Recognition for image with cluttered background become possible with real time detector in 2001 by Viola-Jones [74] Biometric Identification using palmprint identification was proposed in 2003 [80]. With the rapid population explision this task becomes seemingly more challenging. Biometrics based application can now be seen in use around us in everyday life. Government across the world have started to provide their citizens with the biometric identifiers and maintain identity database. These databases are used at airports and other entry points to regulate public movement across borders and establish identity in commercial transactionssingle out suspicious elements. USA, Brazil, Germany, United Kingdom, Iraq, Israel, Australia, New Zealand etc have already started issuing passports containing digitized biometric data like signature, photographs, iris information etc. Many country including India are leading down the same path to maintain the digital records of its population and are in the process of issuing passports and Unique Identification Number with embedded biometrics details.

Biometrics as a solution of user identification and security problems in todays network is believed by a lots of people. Misuse of password, password theft is a big problem in today's network, whether it is human error and in some cases malicionsness. Biometric Technology reduces the scope of human error. That means the case of password lost does not exist.

Biometrics Application can be seen in some high securing buildings and even in some webs. This indicates the importance of the biomerics in the future. It has been in widely used in forsensics applications such as criminal identification and prison security. The biometric technology is rapidly expanding and has a strong potential to be widely accepted in civilian technology. Researchers all over the world working on the biometric technology can be used in the areas like electronic banking, e-commerce. Some companies are working on the implementation of fingerprint authentication system. These days some of the laptop come with the fingerprint authentication system.

With the rapid growth of population and increase in the web technology use of electronic transactions, electronic banking and electronic commerce are becoming one of the most important field in the applications in biometrics. The applications where biometric applications can be practised include credit card security, ATM security, check cashing and fund tansfes online transactions and web access. The use of biometrics will become more widespread in coming years as the technology matures and becomes more trust worthy. Some of these applications have already started using biometrics for person verification. Traditional knowledge-based (password or Personal Identification Number (PIN)) and token-based (passport, driver license, and ID card) identifications can be compromised to fraud because PINs may be forgotten or guessed by an imposter and the tokens may be lost or stolen. Biometrics trait offers a natural and reliable solution to certain aspects of identity management by utilizing fully automated or semi-automated schemes to recognize individuals based on their physiological characteristics. By using biometrics it is possible to establish an identity based on who you are, rather than by what you possess, such as an ID card, or what you remember, such as a password. In some applications, biometrics may be used along with the ID cards and passwords thereby providing an additional level of security. This is known as dual-factor authentication scheme.



(a) (b)

Figure 1.1 Authentication schemes. (a) Traditional schemes use ID cards and passwords to validate individuals and ensure that system resources are accessed by a legitimately enrolled individual. (b) With the advent of biometrics, it is now possible to establish an identity based on who you are rather than by what you possess or what you remember.

Traditional knowledge-based and token-based approaches are unable to satisfy the security requirements of our electronically interconnected information society. Biometrics Identification system such as IAFIS [34] have huge biometric databases. In such a large database, one has to determine the identity of a subject from a large set of users already enrolled in the databases. The identification of a person requires a comparison of the biometric traits to all the traits in the database. Such an action may be necessary for a variety of reasons but the primary intention is to prevent impostors from accessing protected resources. In some cases, the database may be very larger even for a super computer to do one to one

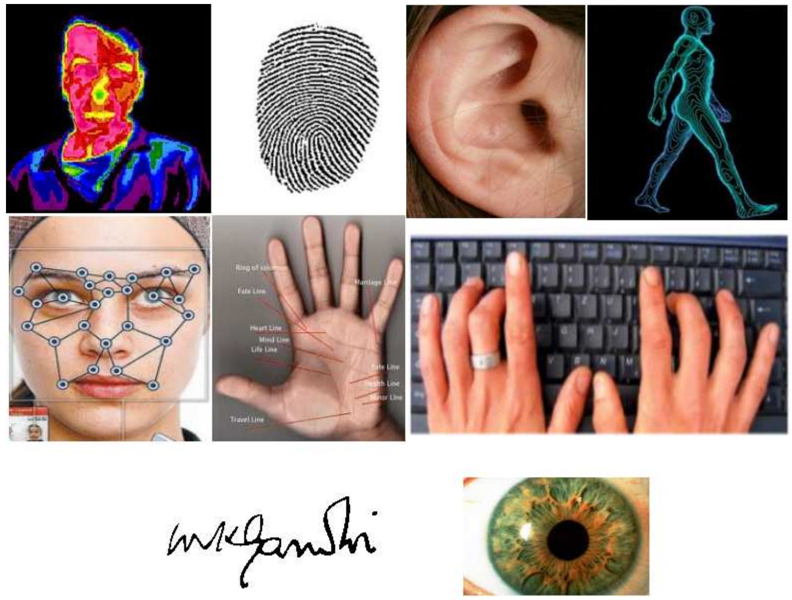


Figure 1.2 Examples of biometric traits that can be used for authenticating an individual. Physical traits include fingerprint, iris, face and hand geometry while behavioral traits include signature, keystroke dynamics and gait.

matching. In such cases identification takes a long time to repond to a query. The current biometrics systems works well with the small database, but it will fail when we have to run it in a larger database, as in the case of Unique Identification. Traditional databases index the records in an alphabetical or numeric order for efficient retrieval. In biometric templates, there is no natural sorting order by which one can sort the biometric records, making indexing a challenging problem. In this thesis We propose a guidelines for the search in biometric databases with the use of filtering.

The introduction to thesis is organized as follows: In this chapter, we present an overview of biometrics. This includes a basic introduction to biometrics, followed by a section explaining the biometric recognition process. We will describe the already existing Indexing process and their short comings. The chapter concludes by giving a detailed motivation to the problem chosen for this thesis, exact problem statement, and thesis contributions.

1.2 Identification vs. Verification

Depending on the application context, a biometric system may operate either in the verification or identification mode (see Figure 1.4,1.5). In the verification mode, the system validates a person's identity by comparing the captured biometric data with her own biometric template(s) stored in the system database. In such a system, an individual who desires to be recognized claims an identity, usually via a PIN, a user name or a smart card, and the system conducts a one-to-one comparison to determine whether the claim is true or not (e.g., "Does this biometric data belong to Atif?"). Verification

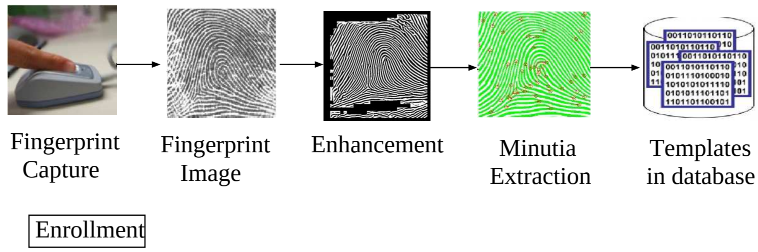


Figure 1.3 Fingerprint Enrolment: First the the image is catured using fingerprint sensor, its enhanced using fingerprint enhancing algorithm, its features (minutiaes position,orientation and its quality) is extracted and its value is stored in the database along with the information about the user.

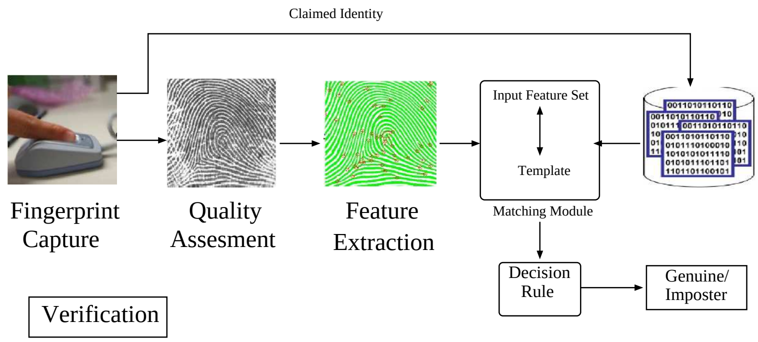


Figure 1.4 Fingerprint Verification: First the the image is catured using fingerprint sensor. The quality assessment module determines if the sensed data can be effectively used by the feature extractor. Its features, (minutiaes position,orientation and its quality) is extracted and its value is calculated and matched with the template of the claimed user.

is typically used for positive recognition, where the aim is to prevent multiple people from using the same identity.

In the identification mode, the system recognizes an individual by searching the templates of all the users in the database for a match. Therefore, the system conducts a one-to-many comparison to establish an individuals identity (or fails if the subject is not enrolled in the system database) without the subject having to claim an identity (e.g., “Whose biometric data is this?”). Identification is a critical component in negative recognition applications where the system establishes whether the person is who she (implicitly or explicitly) denies to be. The purpose of negative recognition is to prevent a single person from using multiple identities. Identification may also be used in positive recognition for convenience (the user is not required to claim an identity). While traditional methods of personal recognition such as passwords, PINs, keys, and tokens may work for positive recognition, negative recognition can only be established through biometrics.

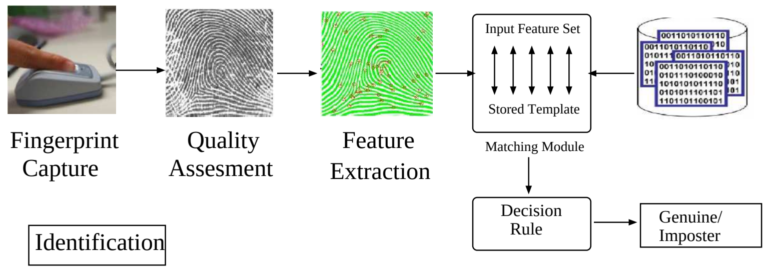


Figure 1.5 Fingerprint Identification: Image is captured, its feature is extracted and its matched with all the templates stored in the database.

1.3 Biometrics Characteristics

Each Biometrics has its pros and cons, therefore, the choice of a biometric trait for a particular application depends on a variety of issues besides its matching performance. Wayman *et al.* [75] have identified seven factors that determine the suitability of a physical or a behavioral trait to be used in a biometric application.

- **Universality** Every individual should have this particular trait.
- **Uniqueness** The trait should be unique for the all population that ever came.
- **Permanence** The biometric trait of an individual should be invariant to time and condition.
- **Measurability** It should be possible to extract the features from the trait.
- **Performance** The recognition accuracy and the resources required to achieve that accuracy should meet the constraints imposed by the application.
- **Acceptability** Individuals in the target population that will utilize the application should be willing to present their biometric trait to the system.
- **Circumvention** The affect of artificats in which the trait of an individual can be affected by fake fingers, mimmicry, iris and capturing devise. This is known as circumvention. An ideal biometric should have minimal circumvention.

Each form of biometrics authentication has its own strength and weakness. No single biometric trait is expected to effectively meet all the requirements (e.g., accuracy, practicality, cost) imposed by all applications (e.g., Digital Rights Management (DRM), access control, welfare distribution). In other words, no biometric is ideal but a number of them are **admissible** Some of the commonly biometric traits are given below:

1. **Face:** Face Recognition is the most common and oldest method used by the human to recognise one another almost as old as human civilisation. The most popular approaches to face recognition

[40] are based on the location and shape of facial attributes, such as the eyes, nose, lips, and chin and their spatial relationships, or the overall analysis of the face image that represents a face as a weighted combination of a number of canonical faces. In order for a facial recognition system to work well in practice, it should automatically detect whether a face is present in the acquired image; locate the face; identify the amount of rotation and recognize the face from a general viewpoint (i.e., from any pose) under different lighting condition.

2. **Fingerprints:** Fingerprints is the strongest of all biometric traits and are used for personal identification for many years. The matching accuracy using fingerprints has been very high [76]. It has been determined that the fingerprints of identical twins are different and so are the prints on each finger of the same person [41]. Today, most fingerprint scanners cost less than $Rs3000$ when ordered in large quantities and the marginal cost of embedding a fingerprint-based biometric in a system (e.g., laptop computer) has become affordable in a large number of applications. Nowadays laptop are coming with fingerprint scanner. The accuracy of the currently available fingerprint recognition systems is adequate for authentication systems in several applications, particularly forensics. The fingerprint for the small section of the society may not be used for the identification, e.g manual workers may have a large number of cuts and bruises on their fingerprints that keep changing. There some cons in using fingerprints as it has traditionally been associated with criminal activities and thus users could be reluctant to adopt this for of biometric authentication

3. **Hand geometry:** Hand geometry recognition systems are based on a number of measurements taken from the human hand, including its shape, size of palm, and the lengths and widths of the fingers and diagonally measurements of palm [11]. Commercial hand geometry-based authentication systems have been installed in hundreds of locations around the world. However, this system can not be deployed on large scale as its the size of features can change with age. The physical size of a hand geometry-based system is large making it difficult to deploy in all the places like laptop.

4. **Palmprints:** The palms of the human hands contain pattern of ridges and valleys much like the fingerprints. The area of the palm is much larger than the area of a finger and, which makes palmprints to be even more distinctive than the fingerprints [80]. Human palms also contain additional distinctive features such as principal lines and wrinkles. These features are easier to capture even with a lower resolution scanner, which would be cheaper. Finally, when using a high-resolution palmprint scanner, all the features of the hand such as geometry, ridge and valley features principal lines, and wrinkles may be combined to build a highly accurate biometric system. The negative thing would be it will be diffcult to deploy in all the places.

5. **Iris:** The iris is the annular region of the eye bounded by the pupil and the sclera (white of the eye) on either side. The visual texture of the circumcentre in iris is formed during fetal development

and stabilizes during the first two years of life. The complex iris texture carries very distinctive information useful for personal recognition [20]. The accuracy and speed of currently deployed iris-based recognition systems is promising and support the feasibility of large-scale identification systems based on iris information. It is easier to fool the system with fake contact lense (see [11]). The hippus movement of the eye may also be used as a measure of liveness for this biometric. Although early iris-based recognition systems required considerable user participation and were expensive, the newer systems have become more user-friendly and cost-effective. It detects the iris and captures it automatically [30].

Some other used biometric characteristics, includes keystroke. This biometric is not expected to be unique to each individual but it may be expected to offer sufficient discriminatory information to permit identity verification [58]. Signature of a person is well known biometrics and is often used in for bank cheque processing system, but it is yet to applied on a automatic system. Voice is another biometrics characteristics. There is some problem with the voice recognition system, like background noise must be controlled. Space required is very large for storing the template. It can be affected by the climate situation,like sore throat, common cold. The physical features of an individuals voice are based on the shape and size of the appendages (e.g., vocal tracts, mouth, nasal cavities, and lips) that are used in the synthesis of the sound. There are two different types of voice recognition system: text-dependent and text independent. Gait refers to the manner in which a person walks, and is one of the few biometric traits that can be used to recognize people at a distance. Based on the study of biometric characteristics its difficult to find a characteristics with all the features. We choose the characteristics based on the situation, level of security needed and costs we can afford.

1.4 Indexing and Classification

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 and in some cases billions of 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. Information about sex, race, age, location and other data related to the individual are available and the portion of the database to be searched can be significantly reduced. These informations are not always accessible (e.g., criminal identification based on latent fingerprints or in case when we are chacking for frauds) and, in the general case, information intrinsic to the biometric samples has to be used for an efficient retrieval. With the advancement of technology, several computers can be run in parallel to retrieve results. 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. Classification referes to a problem in which a class is assigned to fingerprint or palm print. When a probe image comes it class is determined and it searched in database with its own class. The aim of Indexing is to retrieve a small portion of database in order to determince the possible match This improves the reponse time and and enables the implementation of biometrics technology in real world application.

1.5 Existing Methods for Searching in Large Databases

Hao *et al.* [31] shows the use of indexing in large databases. Their used Beacon Guided Search (BGS), tackles the problem of large databases dispersing a multitude of "beacons" in the search space. Despite random bit errors, iris codes from the same eye are more likely to collide with the same beacons than those from different eyes. By counting the number of collisions, BGS shrinks the search range dramatically with a negligible loss of precision. They evaluated this technique using 632,500 iris codes enrolled in the United Arab Emirates (UAE) border control system, showing a substantial improvement in search speed with a negligible loss of accuracy. This is first step step towards indexing of biometric data in a large scale.

Another example of searching in a large database is internet. The internet offers an enormous amount of information in almost every imaginable category. Eric Schmidt, the CEO of Google the world's largest index of the Internet, estimated the size at roughly 5 million terabytes of data. That's over 5 billion gigabytes of data. Schmidt further noted that in its seven years of operations, Google has indexed roughly 200 terabytes of that, or $.004\%$ of the total size [52]. The Indexed Web contains at least 7.74 billion pages as of May 2012 [22] The estimations is based on the numbers of pages indexed by Google, Bing, Yahoo Search Engines.

Even a single website can contain huge collections of data. If visitors have to search these websites without any help it would literally take them hours to find something they where after.

Website indexing concepts developed as indexers, librarians and web managers experimented with different approaches for making the information they were providing on the Internet more accessible. These approaches included: A to Z indexes; displaying the overall structure of the site (information architecture); site maps; and search facilities. Search facilities were sometimes enhanced by the creation of subject metadata ("catalogue cards"), which could be organised in different facets, or displayed visually as well as textually. The tools for creating A to Z indexes have changed over time. Initially, indexers used simple HTML coding to create indexes. Features such as indents and turnaround (wraparound) lines caused difficulties.

The development of HTML Indexer was a major breakthrough in the field of search engines. It provided an effective way to create indexes for websites. They have changed the system and traditional methods of indexing. In one of indexing scheme, an index of all possible query terms is prepared in advance during training phase. Lets take an example of collection of English books in a library. The easiest and the most obvious approach would be to keep track of all words from the English dictionary that appear in each book. On repeating this across all books, we end up with a term-incidence matrix, in which each entry shows if a specific word occurs in a book or not. Figure 1.6 shows a sample term-document identifier

term	1	2	3	4	5	6
the	X	X	X	X	X	X
to	X		X	X	X	X
john		X		X		X
realize	X		X			X
algorithm					X	

Figure 1.6 Term-Incidence Matrix for a sample of English documents taken from [18]

Dictionary		Posting Lists (document identifier, term frequency)					
the	→	1, 9	2, 8	3, 8	4, 5	5, 6	6, 9
to	→	1, 5	3, 1	4, 2	5, 2	6, 6	
john	→	2, 4	4, 1	6, 4			
realize	→	1, 2	3, 1	6, 3			
algorithm	→	5, 3					

Figure 1.7 Illustration of Posting Lists for Example from Figure 1.6([18])

incidence matrix. The collection of documents over which a search engine performs retrieval is referred to as a corpus. This will create a big matrix. For a database of 1 million documents with 100K distinct words, $\backsim 10GB(1M \times 100K)$ will be required to hold the index in matrix form. A total of 4 GB ($1M \times 1000 \times 4$) storage if each document is 1000 words long on average and each value requires 4 bytes to store. Clearly, lot of space is wasted in recording the absence of terms in a document, and hence a much better representation is required to record only the occurrences. [18]

An improved efficient index structure is an inverted index. Its a collection of lists, one per term, recording the documents containing that term [18]. Each item in the list for a term, also referred to as a posting, records the orginal document identifier d, and its corresponding term frequency (TF): $\langle d, tf \rangle$. If 4 bytes are used to encode each posting, a term appearing in 100K documents will result in a posting list of size 100KB to 1MB. We illustrate this in Figure 1.7 for the same example as in Figure 1.6.

Another interesting problem in the same scale includes searching image from in a internet. Image search is a specialized data search used to find images. User can enter query terms such as keyword, image file, link to the image. for images, a user may provide query terms such as keyword, image file/link, and the system will return images “similar” to the query. The similarity used for search criteria could be meta tags, color distribution in images, region/shape attributes, etc.

- **Image-meta Search** search of images based on associated metadata such as keywords, text, which are linked to the image.

- **Content-based image retrieval** Content Based image retrieval provides result based on the content of the query image. It uses various application of computer vision to retrieve results. It avoids the use of text to retrieve results and its purely based on colors shapes, textures.

The most misunderstanding when it comes to image search based on meta data is that the most people thinks that search is based on detecting information in the image itself. But most image search works as other search engines. The metadata of the image is indexed and stored in a large database and when a search query is performed the image search engine looks up the index, and queries are matched with the stored information. Its like an html indexer which returns image. The results are presented in order of relevancy. The usefulness of the result

"Content-based" means that the search will analyze the actual contents of the image like colors key points inside the image, rather than the text data such as keywords, tags, and/or descriptions associated with the image. The term 'content' in this context might refer to colors, shapes, textures, or any other information that can be derived from the image itself. CBIR is desirable because most web based image search engines rely purely on metadata and this creates a large variability in the results. Google Image has started content based search. Tineye is another content based image search. Concept-based image indexing, also variably named as "description-based" or "text-based" image indexing/retrieval, refers to retrieval from text-based indexing of images that may employ keywords, subject headings, captions, or natural language text [15]

Chen *et al.* [16] have shown that hierarchical trees and pyramids are very effective for both searching and browsing large databases of images. In [3], Weiss have develop efficient image search and scene matching techniques that are fast and require very little memory, enabling their use on standard hardware or even on handheld devices. Their approach uses the Gist descriptor (a real valued vector that describes orientation energies at different scales and orientations within an image) to a compact binary code, with a few hundred bits per image. Sift features are also in use for Content based matching of image in a large databases. Large databases of biological data is processed to obtain information about biological structures of different species [9]. In computer vision and pattern recognition, problems like face recognition [56], body pose estimation [68], optical character recognition [5] require processing on enormous amount of data.

The large scale problem discussed above either follows a natural sorting order(in case of internet indexing) or in most cases it doesnt affect the result much if there is some false negative. Incase of biometrics, the intraclass difference is very low, and interclass difference is almost fixed between any two samples belongs to different class. Size of the feature vector is very large, which increases the computational time between two samples. There may be chances that part of the probe sample is missing(like in the case of fingerprint where some minutiae may be missing). Size of the biometric database is huge, which keeps on increasing everyday. Absence of a reliable and fast indexing method shows the relevance of the problem.

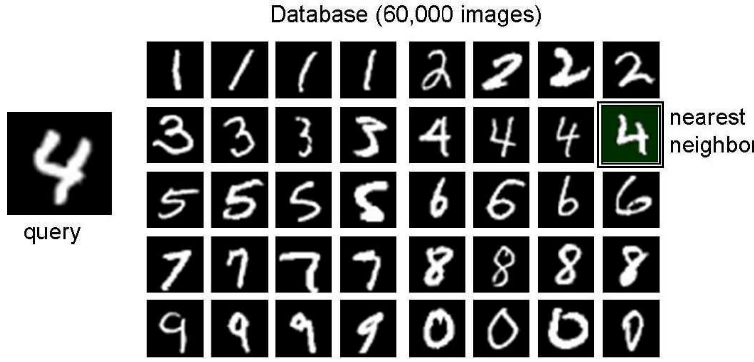


Figure 1.8 Handwritten digit recognition using a nearest neighbor classifier and the MNIST database of 60,000 training images [5]. Given a query image that we want to classify, the system retrieves the nearest neighbor of the query in the database, and assigns the class of the retrieved nearest neighbor to the query image.

1.6 Problem Statement and Thesis Overview

The problem we deal in this thesis, is the automatic generation of filtering pipeline that reduces the search space, for a given query image, with minimal false non matches. We are proposing the use of cascaded random projection.

In chapter 2 We have explained the history of biometrics, indexing and classification, a brief review of Random Projection, its advantage over earlier form of dimensionality reduction, its cost effectiveness, its accuracy. We have explained some of the state of the art indexing methods in different biometrics. In chapter 3, use of random projection in palmprint and iris for cascaded filtering. In chapter 4, cascaded filtering of fingerprints data, using random projection is explained. We conclude thesis with a conclusion and some future work in the field of biometric indexing.

1.7 Summary

With the rapid advancements in the field of communications, computer networking and transportation, along with increased concerns about identity fraud and national security, has resulted in a solid need for reliable and efficient identity management schemes. Identity management includes creation, maintenance and removal of identities along the guarantee of protection from impostor does not fraudulently gain privileges associated with a legitimately enrolled individual. Traditional techniques based on passwords and tokens are limited in their ability to address issues such as negative recognition and non-repudiation. Password can be compromised. Tokens can be compromised. Biometric systems use the physical and behavioral characteristics individual to establish an identity which makes it more secure as its difficult fool and remembering password is not an issue.

The deployment of biometrics in civilian and government applications has raised questions related to the privacy of an enrolled individual [21]. Specifically, questions such as the biometric data be used to track people thereby violating their right to privacy. The acquired biometric data be used only for the intended purpose of verifying, or will it be used for some other functions? Individual's financial and social profile should be secure, possible outcome if the system fails to correctly identify biometric data have advocated raise several concerns about the use of biometric solutions in large-scale applications. Even if these problem are resolved, the implementations of biometrics system on a national level is still a challenge. The indexing of such a large database is a challenge and still a lots of work can be done in the field.

Chapter 2

Previous Work

In the last chapter, we present an overview of biometrics, indexing of biometrics. Some large scale problem. in detail. Here, we discuss about the existing works in the field of indexing of fingerprints, iris and palmprint recognition and some indexing method in detail. We start with a previous works in the field of Random Projection, its cost effectiveness ability over traditonal type of dimensional reduction method in section 2.1. In section 2.2 a comparison between classification and indexing. The Classification based approach for indexing is discussed in section 2.3. In the subsections we have explained the classification in fingerprints, Henry's classification and palmprint classification. In the section 2.4 we explained Indexing works in the biometrics and explained fingerprints and iris indexing. In section 2.4.4 we have give a brief overview of palmprint indexing. We explained the pyramid indexing in section 2.5. We conclude the chapter with a summary and overview of the next chapter.

2.1 Random Projection

The Nearest Neighbour(NN) Search problem, is a major problem in various fields of computer science. Basically there are two sides of the problem: Exact NN and approximate NN. The problem is stated as: Given a set of points P a high-dimensional space, construct a data structure which given query point q finds the point in P closest to q(for exact NN). or a close approximation to the nearest point of q(for approximate NN). The problem stated above is of significant importance to field of pattern recognition, searching in multimedia data, dimensionality reduction [28], computational statistics [23], data mining etc. Many of these applications involve data sets that are very large in dimension and size. Moreover the dimensionality of the data points can be in the order of hundreds or thousands. Both of these factors make it a challenging computational problem in computer science.

Random Projection has been used in past for dimensionality reduction. This is a technique of mapping a number of points in a high-dimensional space into a low dimensional space with the property that the Euclidean distance of any two points is approximately preserved through the projection, where the high dimensionality of the data would otherwise lead to heavy computations. A promising dimensionality reduction method for a use in pattern recognition is random projection. In Sulic *et al.* [70]

Biometric Trait	Iris	Signature	Ear	Face	Combined Features
FRR	2%	2.1%	1.8%	3.33%	0.66%

Table 2.1 Result on indexing with KD tree on different Biometric Traits

the performance of the random projection method which can be used in embedded cameras is shown. Random projection is compared to Principal Component Analysis(PCA) in the terms of recognition efficiency on image data set. Unlike PCA, it does not depend on a particular training data set. Unlike Discrete Cosine Transform (DCT) or Discrete Fourier Transform (DFT) its basis vectors do not exhibit particular frequency or phase properties. Results in [70] shows good performance of the random projection in comparison to the PCA even without explicit normalization of transformation subspace. Furthermore, it represents a computationally simple and effective method that preserves the structure of the data without significant distortion [29]. It preserves for example volumes and affine distances [50] or the structure of data (e.g. clustering) [19].

In [25] random projection is used for a cluster ensemble approach. In this approach multiple runs of clustering are performed and the results are aggregated to form an $n \times n$ similarity matrix. Here n is the number of instances. A clustering algorithm is then applied to the matrix to produce the final cluster. Dasgupta *et al.* [19] showed that random projection can change the shape of highly eccentric clusters to be more spherical. In [32], it showed the random projection method can be used in conjunction with standard algorithms with virtually no degradation in performance. Random projections can been shown to result in both significant computational savings and provably good performance. Bingham *et al.* in [7] showed the use of Random Projection for dimesnsional reduction in noisy and noiseless images. Goel *et al.* [29] uses the Random Projection for face recognition. Their experiments representing faces shows that although random projections represents in a random, low-dimensional subspace, its overall performance is comparable to that of PCA with a lower computational requirements and being data independent.

However, one of the major drawback of random projection is that it is highly unstable different random projections may lead to radically different clustering results.

2.2 Multidimensional Indexing in Biometrics

Jayaraman et al. [43] explained the usage of multidimensional indexing in biometrics with B+ trees. The features are projected on to a low dimensional subspace defined by PCA. The indexing is then performed on this low dimensional Eigen space using B+ trees. In most of the application where multidimensional techniques has been used, either B+ trees or KD-Trees has been used to index. Result is shown in table 2.1.

Depends on the different value of k chosen the penetration rate varies from 3% to 12% on the combination of featured from iris, signature, ear and face.

2.3 Classification and Indexing

Classification based indexing means assigning a class label to the probe which is then compared with the gallery template which belongs to the same class as the probe. The main issue with the classification technique is that the number of classes is small and templates are unevenly distributed among them. This problem is addressed in using sub classification and continuos classification. Another critical issue is that the accuracy of the overall system is limited to that of the classifier used for indexing, which is too low to useful.

2.4 Classification in Biometrics

2.4.1 Fingerprint classification

Classification as defined above is assigning a label to probe. Fingerprint classification is generally based on global features, such as global ridge structure and singular points (Core and Deltas). Fingerprint classification is a challenging pattern recognition problem due to the small inter-class variability and the large intra-class variability in the fingerprint patterns. Moreover, fingerprint images often contain noise, in some cases only contains a part of image, which makes the classification task even more difficult.

The Henry Classification System, the first British fingerprint files in London, written by Edward Henry more than 100 years ago, was a precursor to the fingerprint classification system that was used by the FBI for many years.

2.4.2 Henry Classification System for fingerprints

We will start with how fingerprint classification progress over the years. Purkinje in [57], proposed first fingerprint classification rules in 1823. He classified fingerprints into nine categories (transverse curve, central longitudinal stria, oblique stripe, oblique loop, almond whorl, spiral whorl, ellipse, circle, and double whorl) according to the global ridge configurations. Francis Galton performed the first detailed scientific work on fingerprint classification. He divided the fingerprints into three major classes (arch, loop, and whorl) and further divided each category into subcategories [26]. During the same period, a police official from Argentina, Juan Vucetich from developed a different system of classification. This system is still used in many Spanish-speaking countries. Edward Henry, a British police officer in Bengal in 1900 refined Galtons classification by increasing the number of classes [33]. The GaltonHenry classification scheme was adopted in several countries. Most of the classification schemes currently used by law enforcement agencies worldwide are variants of the GaltonHenry classification scheme.

FBI follows this system of classification and used eight different classes for fingerprints: radial loop, ulnar loop, double loop, central pocket loop, plain arch, tented arch, plain whorl, and accidental [72]. As a result, most automatic system reduces the number of classes to five. The Henry Classification System allows for categorization of ten-print fingerprint records into primary groupings based on fingerprint pattern types. This system reduces the effort necessary to search large numbers of fingerprint records by classifying fingerprint records according to gross physiological characteristics.

To reduce the search time and computational complexity, it is desirable to classify these fingerprints in an accurate and consistent manner such that the input fingerprint needs to be matched with only a subset of the fingerprints in the database. When an input fingerprint comes it is matched with on of the pre specified fingerprints to determine which class the probe fingerprint belongs. Then this input fingerprint image is matched with all the fingerprint in the database. The population distribution of the occurence of the fingerprints are approximately 33% , 36%, 17%, 6%, and 8% for whorl, right loop, left loop, arch, and tented arch, respectively [72] according to the survey by FBI. Sometimes it happens with the automated system that we get two classes for a query template. We can see with the population distribution that for two classes, whorl and right loop, it covers about 70% of the whole population. Such occurence reduces overall effectiveness of classification based indexing. This classification is not in much use today, as it was used few decades ago.

2.4.3 Palmprint Classification

Palmprint classification provides an important indexing mechanism in a palmprint database. As discussed in chapter 1 it is one of the important biometric modalities. An accurate and consistent classification can greatly reduce palmprint matching time for a large database. Xiangqian *et al.* in [78] principal lines of the palmprint is defined using their position and orientation and thickness. After that a set of directional line detectors is algorithm is run. Then the potential beginnings of line initials of the principal lines are extracted. After that, based on these line initials, a recursive process is applied to extract the principal lines. Palmprints are classified into six categories according to the number of the principal lines and the number of their intersections. The proportions of these six categories (1-6) in their database containing 13,800 samples are 0.36%, 1.23%, 2.83%, 11.81%, 78.12% and 5.65%, respectively [24]. This approach is able to classify palmprints with an accuracy of 96.03%

In Fang *et al.* [24], it is shown a palmprint classification algorithm which is able to classify palmprints into ten evenly-distributed categories (1, 2, 3, 4, 6, A, B, C, D, and E). In palmprint biometric system, 78% of the population falls in one category.

(a) Double Whorl

(b) Whorl

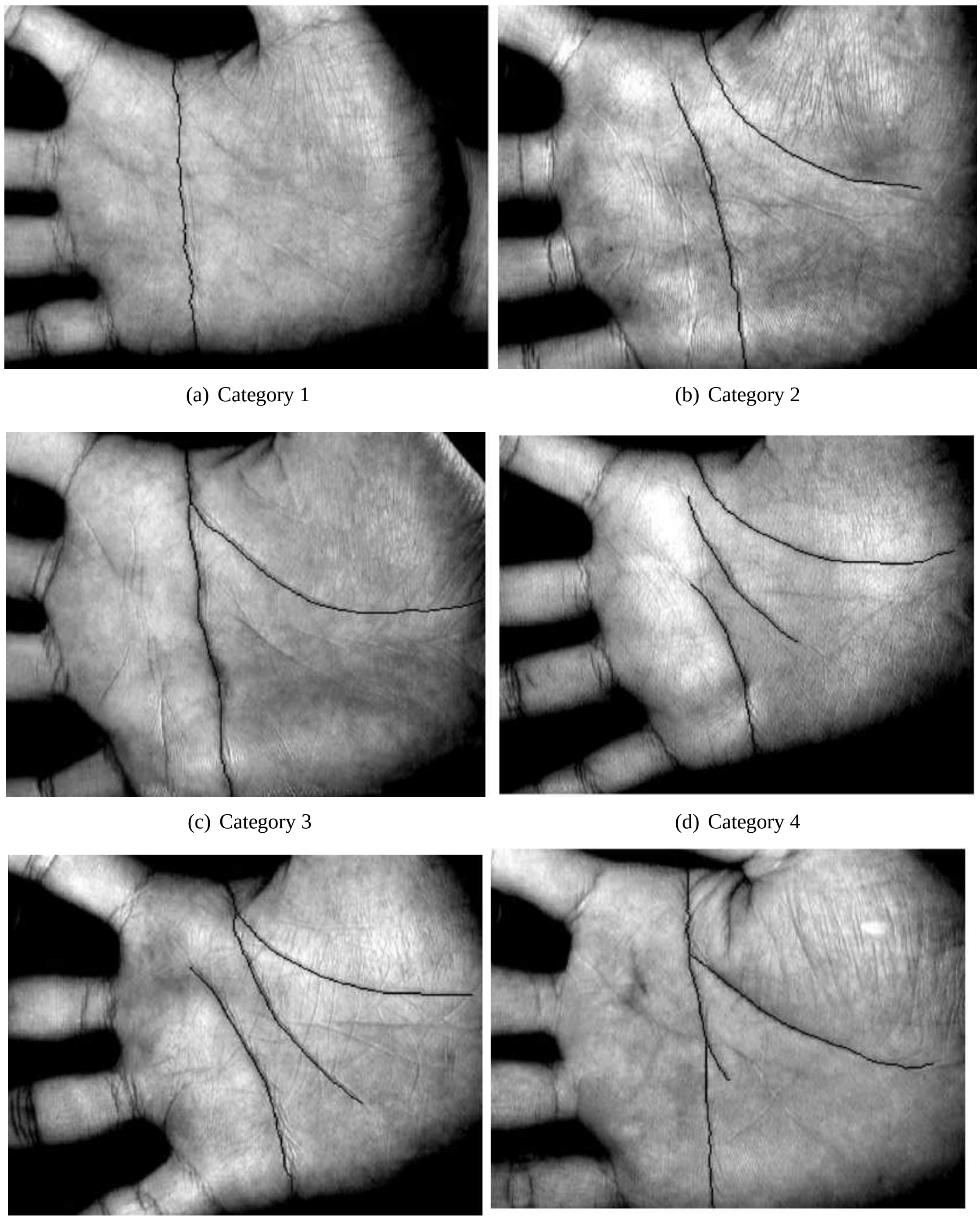
(c) Right loop

(d) Left loop

(e) Arch

(f) Tented Arch

Figure 2.1 6 different types of fingerprints that are used in academics and indstries for classification



(a) Category 1 (b) Category 2

(c) Category 3 (d) Category 4

(e) Category 5 (f) Category 6

Figure 2.2 Examples of each palmprint category.

2.5 Indexing in Biometrics

2.5.1 Fingerprints Indexing

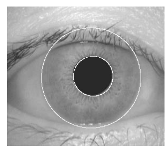
Germain *et al.* in [27] used the triangulation of mintutiae points, length of each side, local orientaion and ridge count between two vertices for the indexing of fingerprints. False positive rate on a 10 person database is 10%. FPR on 100 person database is 63%. 32 disks were distributed over 8 node IBM SP2 system, could search a database of 10 million fingerprints in 70 seconds. Bhanu *et al.* uses triangulation of minutia [6]. He used the maximum of the three sides, median and minimum angles, triangle handedness, type and direction of the minutiae selected, ridge count and minutia density. On NIST-4 database the Correct Index Performance (CIP) is 85% with the verification of his method was limited to the 10% of the database. Bebis *et al.* in [4] uses the delaunay triangulation (see Figure 2.3) method of minutiae points. The other feature used was the ratio of the maximum to the minime length of the triangle and the cosines of the two smallest sides. In case of 3 imprints per person in the training set, average correct matching rate is 86.56%. and the average false negative matching rate is 13.36%. Arun Ross and Mukherjee in [66] uses Delaunay Triangulation to extract the features. He choses the largest angle, ratio of the square of the perimeter and area of the triangle and ratio of the longest to the smallest side as the feature. The penetration was 51.4% for 100% hit rate. and 39.52% for 80% hit rate.

Liu *et al.* [49] shows indexing based on the Singular point correlation. He proposes the continuous fingerprint indexing method based on location, direction estimation correlation of fingerpeint singular points. In 2006 he presented fingerprint indexing method based on LAS registration [48]. The average search space was 2.34% of the total database if the size of the testing and training is same of FVC 2000 dataset. Cappelli *et al.* in [13] shows the use of minutia cylinder code for fingerprint indexing. A Locality-Sensitive Hashing (LSH) scheme has been designed relying on Minutiae Cylinder-Code (MCC), which proved to be very effective in mapping a minutiae-based representation (position/ angle only) into a set of fixed-length transformation-invariant binary vectors. MCC has been used in fingerprint matching in [12]. MCC is a novel representation based on 3D data structures (called cylinders), built from minutiae distances and angles. The cylinders can be created starting from a subset of the mandatory features (minutiae position and direction) defined by standards like ISO/IEC 19794-2 (2005). They have demonstrate the feasibility of obtaining a very effective (and interoperable) fingerprint recognition implementation for light architectures.

2.5.2 Iris Indexing

In iris recognition is based on the texture content of iris is used to extract features, which are used for iris recognition [37]. Ross *et al.* [67] shows the use of iris codes for indexing. Iris template generation involves the two stages, the first stage involves iris segmentation, where the iris is localised and isolated from the other structures in the vicinity, second stage geometric normalization is done, where the annular structure of the iris is mapped to the polar domain via an “unwrapping” procedure resulting

Figure 2.3 Delauney Triangulation method used in different paper for fingerprint Indexing(Figure taken from [66])



(a) IRIS with a circle

(b) Unwrapped iris template

Figure 2.4 Segmented iris and its unwrapped template

in a rectangular entity. Finally the feature extraction, where this rectangular entity is projected onto a Gabor wavelet and the resulting phasor information quantized into an IrisCode. See Figure 2.4. Puhan *et al.* uses the iris color for database indexing method [63]. Mehrotra *et al.* [53] used energy hostogram of DCT subbands. where he used histogram of dct subbands for indexing.

In this method iris is indexed using energy histogram. Iris image is first normalised and then is divided into subbands using multiresolution DCT transformation. Histogram is formed for all the the images in the database, which is divided into fixed bins to group the images having similar values. A detail work is given in [53].

2.5.3 Palmprint Indexing

Not much work has been performed in the area of palmprint indexing. Latest is the work by Yang *et al.* in [79] in 2011. Palmprint is aligned during the offline training by registering its orientation field with respect to a set of reference orientation fields. These orientation fields are obtained by clustering training palmprint orientation fields. Indexing is based on comparing ridge orientation fields and ridge density maps, which is much faster than minutiae matching. Algorithm proposed in [79] achieved an error rate of 1% at a penetration rate of 2.25% on a palmprint database consisting of 13,416 palmprints. It takes only 0.22 seconds to retrieve the results.

2.6 Pyramid Indexing

One of the common indexing technique is Pyramid indexing. It works for relatively small dimensional datasets. The key idea is to divide the d dimensional data space first into 2d pyramids sharing the center point of the space as the top as seen in 2.5,2.6. It involves the normalisation of the data values to lie between 0 and 1. Subsequently, each of the single pyramid is cut into slices parallel to the base of the pyramid that forms the data pages. Such a partition strategy yields a mapping from the d-dimensional space to a 1-dimensional space. B+ tree is then applied to index these one dimensional transformed data. The pyramid works well when featue vector size is less [54]. But when the size increases to few thousands, it fails to give any satisfactory result.

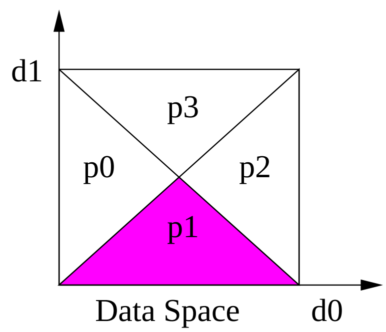


Figure 2.5 Partitioning 2-d space into Pyramids

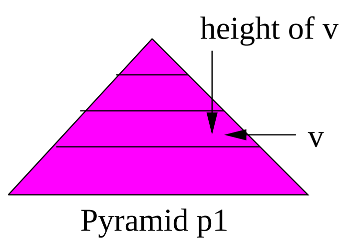


Figure 2.6 Height of point v in Pyramid P1

2.7 Summary

In this chapter, we have seen that random projection can be applied in other fields of computer vision, its cheap computation cost. We have seen all the important indexing process that are used in the biometrics system. The problem of indexing in a large databases a well known problem in computer science. The current indexing method will fail if the size of datasets is over 1 billion. Most of the indexing method that are in use today are either hand crafted or works fine when we have small datasets. With larger size of the feature vector, indexing becomes difficult to perform. We have seen the problem with the classification of fingerprints and palmprints. Huge population of fingerprints and palmprints falls within 3 classes, this reduces the significance of classification. We discuss about the difference between classification and indexing. Random Projection, a novel concept, inspite of compuationally cheap, has never been used in the field of biometrics as a weak classifier. In the next two chapters we discuss about the indexing of biometrics data using random projection. In Chapter 3 we will discuss about the indexing of palmprint using random projection. In chapter 4 we will discuss about the indexing of fingerprints using random projection. We conclude the thesis with a conlusion and future work.

Chapter 3

Cascaded Filtering using Random Projections

We now describe the process of creation of the filtering pipeline using random projections. We also compare the results with the use of non-random projections such as PCA and LDA, along with experimental results on Palmprint and Iris datasets.

3.1 Introduction

Approaches to reducing the search time falls into two categories: indexing and filtering. Indexing, as mentioned before, classifies a probe as belonging to a specific class (or a few classes), and uses only that part of the dataset from the same class for explicit comparisons. The process is extremely quick as the time required for classification of the probe is independent of the database size. However approach assumes that the biometric trait can be partitioned into mutually exclusive set of classes and classification into these classes is accurate. Filtering approaches relaxes this assumption and uses a simple light-weight matcher to compare the probe against each entry in the database. All samples that are potential candidates from this matching process is passed on to the next stage for further comparisons.

We note that the ideal feature representation of strong biometric trait used for identification is not well-suited for indexing as the inter-class distances tend to be close to each other as evidenced by low variance of the imposter distribution. Similarly, the variable length of feature representation and the comparison mechanisms used in practice for strong biometric traits makes it too heavy for use in a filtering process. As supported by experimental evidences, a direct approach to indexing biometric data such as the use of indexing structures like KD-Trees on the feature representations of a strong biometric does not yield satisfactory results. To overcome these difficulties, researchers and biometric practitioners have proposed a variety of features and matching strategies, often tailored to a biometric trait for the purpose of indexing and filtering. As we are using the indexing or filtering stage as a precursor to explicit matching, we would like to keep the False Non-Identification Rate (FNIR), very close to zero, while pruning the database as much as possible. FNIR indicates the probability that a probe with a matching record in the database would return a no-match after the entire identification process.

Automatic classification of fingerprints into the Henry classes was explored by Jain *et al.* [38], yielding a system with 12.4% FRR. A similar work by Ratha *et al.* [65] yielded a False reject rate(FRR) of 10% with search space pruned to 25% of the original database. In an experiment conducted by Cappelli *et al.* [64] on NIST Special Database 4, it was shown that the distribution of Fingerprint population was non-uniform with 2 of the 5 Henry classes they considered holding nearly 65% of the population. Note that the FNIR (corresponds to FRR in this case) is too high for most practical purposes and often one has to search more than one bin in the database for every probe. This further reduces the effectiveness of the method.

The pyramid indexing [55] technique tries to map a feature vector into one of the pyramids centered at the mid point of the feature range. The index of the pyramid and the location of the probe within the pyramid helps to reduce the search space to points within a few pyramids in the database. The authors report considerable success with this technique, with a database pruned to 8.86% of original size with 0% FNIR in case of hand geometry. Unfortunately the method performs poorly with larger feature vectors such as Gabor responses of IRIS images. Mehrotra *et al.* [53] proposed the use of ordered DCT coefficients for indexing a dataset of IRIS images. The authors were able to prune the database to around 2.6% with an FNIR of 35.6%. The method is sensitive to the location and orientation of the samples and does not work well with other modalities such as palmprints or fingerprints.

For palmprints, Zhang *et al.* [80] proposed the use of high-level textural information to filter out a set of possible candidates for fine-grained matching using interest points. Hierarchical identification of palmprint, where a Hough transform of the principal lines is used as a feature for filtering was proposed by Li and Leung [24]. Local information extracted from line-based Hausdorff Distance (LHD) is used for further fine-level identification.

In short, we note that the feature representations and the indexing and filtering schemes developed are often tailored for a specific biometric modality. In this work, we explore the use of random linear projections as a generic method for deriving features from a given feature representation of a strong biometric for the purpose of filtering. We also propose a cascaded window based filtering scheme that would be applicable to such feature representations in an efficient manner.

3.1.1 Random Projections

The use of linear projections to reduce the dimensionality of a dataset is a well explored topic. Approaches such as Principal Component Analysis and Linear Discriminant Analysis try to find a set of projections for a given dataset that would maximize a specific objective function. In other problems such as unsupervised learning, the objective function is either not defined or cannot be optimized analytically. The distance preserving nature of linear projections into random subspaces were explored by Johnson and Lindenstrauss [45] in 1984 (JL Theorem), who showed that random projections preserve the structure of high dimensional data well in lower dimensions. Specifically, the distortion in distances, when mapping n p-dimensional points into a q-dimensional random subspace, where $q \geq O(\log(n)/\epsilon^2)$ is less than a factor of $1+\epsilon$. The method of random projections have been proven to be useful in a variety of practical applications such as dimensionality reduction, density estimation [19], data clustering [25], nearest neighbor search [47, 36], document classification [62], etc.

Random projections have also been used in biometric verification to derive lower dimensional feature representations of modalities such as face [29] and to derive cancelable representations using Multi-space Random Projections [44]. In this work, we explore the use of single random projects as weak classifiers that can act as a filtering stage for efficient biometric identification. We employ each projection as an independent filter in a cascaded fashion [74] to achieve efficient and flexible filtering. The use of cascades as a method for improving efficiency of matching for iris was also suggested in [71].

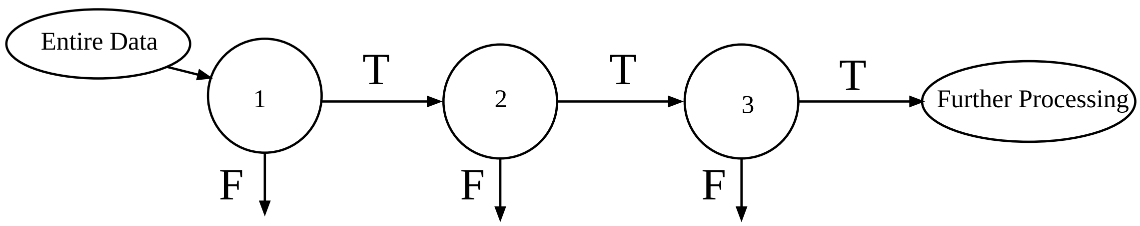


Figure 3.1 Cascading Approach: A large number of weak classifier is used to remove the data which does not belong to a probe sample at each stage.

3.1.2 Principal Component and Linear Discriminant Analysis

An alternative to random projection is to employ projections that maximize certain properties that are suitable to achieve high levels of filtering. The most common approaches to achieve this are Principal Component Analysis or PCA and Fischer Linear Discriminant Analysis (LDA).

Principal Component Analysis (PCA) preserves dimensions with maximum variance for the given data points (Y) and hence are potential candidates for projection for filtering. The first principal component, w_1 is obtained as:

$$w_1 = argmax_{||w||=1} Var\{Y^T w\} \tag{3.1}$$

While PCA is good for minimizing the error in representation of data in low dimensions, it does not promote the separation of classes in the projected subspace. This negatively affects the filtering performance on each projection. Linear Discriminant Analysis (LDA) on the other hand utilizes the class labels of the data and tries to maximize the ratio of between-class variance to the within-class variance in any particular data set, thereby guaranteeing maximal separability. The resulting projection may be used as a linear classifier, or, more commonly, for dimensionality reduction before later classification. LDA considers maximizing the following objective:

$$J(w) = \frac{w^T S_B w}{w^T S_W w} \tag{3.2}$$

where S_B is the "between classes scatter matrix" and S_W is the "within classes scatter matrix". Note that due to the fact that scatter matrices are proportional to the covariance matrices we could have defined

J using covariance matrices the proportionality constant would have no effect on the solution. The definitions of the scatter matrices are:

$$S_B = \sum_c (\mu_c - \bar{x})(\mu_c - \bar{x})^T \tag{3.3}$$

$$S_W = \sum \sum_c (\mu_c - \bar{x})(\mu_c - \bar{x})^T \tag{3.4}$$

where $\bar{x}$ is the overall mean of the data-cases. Oftentimes you will see that for 2 classes S_B is defined as $S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$. This is the scatter of class 1 with with respect to the scatter of class 2 and hence corresponds to computing the scatter relative to a different vector. By using the general transformation rule for scatter matrices:

$$S_{\mu+\nu} = S_\mu + N\nu\nu + 2N\nu(\mu - \nu)^T \tag{3.5}$$

with $S_\mu = \sum (x_i - \mu)(x_i - \mu)^T$, we can deduce that the only difference is a constant shift not depending on any relative distances between point. It will therefore have no impact on the final solution.

The objective function for LDA, the ratio of between-class to within class scatter seems to suit the filtering process, where we would like to have samples of the same class near the query and others, farther away from it. We will explore the relative merits of the approach in the experiments.

3.2 Filtering with Projections

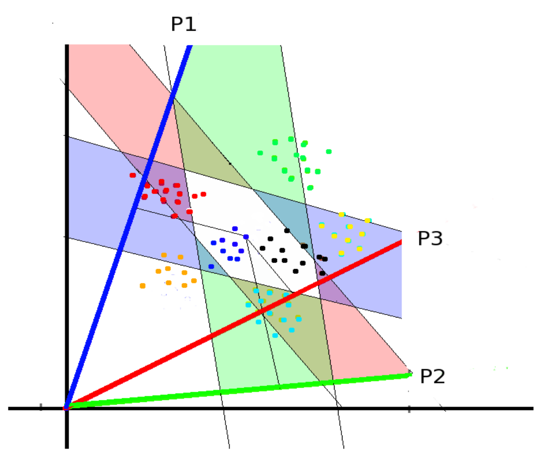


Figure 3.2 Cascading random projections: P1, P2 and P3 are three projections used in a sequence. Samples that are not falling within a window of the probe are removed at each stage.

We consider each projection as a weak but efficient representation of the biometric dataset. Matching against a dataset, where each sample is represented as a scalar is extremely efficient. If the samples of each class are clustered in the projected space, it is reasonable to assume that samples of the same class will be within a window of the probe in the projected space. At each stage, we discard the samples that are outside the window. Figure 3.2 shows the result of projection of a set of two dimensional samples on to three projections and discarding the samples that lie outside a window. The white polygon in the middle represents the samples that are selected from the cascaded filter.

$CandidateList \Leftarrow$ {All templates in gallery}
for each projection P_i **do**
  Retrieve projected values for $CandidateList$ for P_i
  Find the window around the projection of probe on P_i
  Remove templates outside window in $CandidateList$
end for
Return $CandidateList$

Algorithm 1: Computing Candidate list for a probe.

Although the final set of samples that are selected are independent of the order of projections, the efficiency of the cascade is clearly dependent on it. If we use projections that remove large number of impostor samples at initial stages of the cascade, the number of comparisons at later stages of the cascade can be minimized. If the projection preserves the intra-class similarity in the projected space as compared to the inter-class variations, then we can use a small window that would reject a large number of impostors without losing any genuine samples. The property that we like to maximize is hence close but not identical to the Fisher criterion, the ratio of between-class scatter to within-class scatter (S_B/S_w).

After each projection, the data outside the filter window around the probe is analyzed. If the data is from a different class (person), we call it a correct reject (rejected correctly) and if the data is from the same class as the probe, we call it a false reject (rejected falsely). The fitness or goodness of a projection i with a window W may be calculated using the following:

$$c_i = \frac{\displaystyle\sum_{j \notin W} \neg S(j)}{\sum_j \neg S(j)} \tag{3.6}$$

$$f_i = \frac{\displaystyle\sum_{j \notin W} S(j)}{\sum_j S(j)}, \tag{3.7}$$

where $S(j)$ is and indicator variable that takes a value 1, when j is of the same class as the probe and 0 otherwise. N is the total number of samples. The score of the i^{th} projection is defined as the ratio:

$$Score_i = \frac{c_i}{1 + f_i}. \tag{3.8}$$

We note that the definition of this objective function does not yield to an analytic formulation of a minimization problem to find the optimal set of projections. The use of the Fisher criterion will give us the most discriminating set of basis vectors. However, as we note from Figure 3.2, the use of additional projections over a basis set of vectors can further improve the filtration process. To address this problem, we start with a large number of random projections, and select those which maximizes the above criterion function. One could also include the discriminating basis vectors along with the random projection before the selection process. Section 3.4 compares the use of LDA vectors as a projection basis as opposed to random projections, and the effect of its combination.

3.2.1 Advantages of Random Projections

The use of random projections allow us to deal with a variety of problems encountered in other linear projection estimation techniques. As noted before, according to the Johnson-Lindenstrauss lemma [45], a random subspace of dimensionality $O(logn)$ can effectively represent n samples in any high dimensional feature space. Moreover, the use of random projections make the resulting representation to be independent of the training data, and hence addition of new data does not require changes to the random basis. We partially negate this advantage by choosing a subset of the random projections that best filter the training data. One can also produce any number of projections as desired unlike methods such as PCA or LDA that are limited by the rank of the covariance matrix or the number of classes.

Avoiding matrix inversions that are required in the computation of other linear projection methods makes the computation more numerically stable and widely applicable. The training process is also relatively less expensive.

3.3 Implementation Details and Challenges

The first set of experiments on filtering are done on a dataset of palmprint images. As we are using a fixed length feature vector for projection, we need to ensure that the features in the same position of different vector correspond to each other. This makes the process of aligning and cropping palmprints from images, critical. Experimental results for filtering are also reported on a database of Iris images.

3.3.1 Preprocessing

It is important to define a coordinate system that is used to align different palmprint images for matching. To extract the central part of a palmprint, for reliable feature measurements, we use the gaps between the fingers as reference points to determine a coordinate system. The five major steps (see Fig. 3.3) in processing the image are:

Step 1: A lowpass filter, $L(u, v)$ is applied, such as Gaussian smoothing, to the original image, $O(x, y)$. A threshold, Tp, is used to convert the convolved image to a binary image, $B(x, y)$, as shown in Fig. 3.3(b).

Step 2: The boundary is obtained, $(F_i x_j, F_i y_j)(i = 1, 2)$, between the fingers using a boundary tracking algorithm (see Fig. 3.3(c)). The boundary of the gap between the ring and middle fingers is not useful for the following processeing. So its not extracted.

Step 3: The tangent of the two gaps is computed. Let $(x1, y1)$ and $(x2, y2)$ be any points on $(F1xj, F1yj)$ and $(F2xj, F2yj)$, respectively. If the line $(y = mx+c)$ passing though these two points satisfies the inequality, $Fiyj = mF_ix_j + c$, for all i and j (see Fig. 3.3(d)), then the line $(y = mx + c)$ is considered to be the tangent of the two gaps.

Step 4: $(x1, y1)$ and $(x2, y2)$ is aligned to get the Y-axis of the palmprint coordinate system, and use a line passing through the midpoint of these two points, which is perpendicular to the Y-axis, to determine the origin of the coordinate system (see Fig. 3.3(d)).

Step 5: Then we crop the central region based on the coordinate system The sub-image is located at a certain area of the palmprint image for feature extraction (see Figs. 3.3(e)-3.3(f)).

3.3.2 Feature Extraction

As mentioned before, a palmprint can be represented by some line features from a low-resolution image. Algorithms such as the stack filter [77] are able to extract the principal lines. However, these principal lines are not sufficient to represent the uniqueness of each individuals palmprint because different people may have similar principal lines in their palmprints. Fig. 3.4 demonstrates this problem by showing nine different palmprint samples that have similar principal lines. In addition, some palmprint images do not have clear wrinkles (see Fig. 3.5). As a result, we try to extract texture features from low-resolution palmprint images, and we propose a 2-D Gabor phase coding scheme for palmprint representation, which has been used for iris recognition [37].

The circular Gabor filter is an effective tool for texture analysis [37], and has the following general form

$$G(x, y, \theta, u, \sigma) = \frac{1}{2\pi\sigma^2} exp \left\{ -\frac{x^2 + y^2}{2\sigma^2} \right\} exp\{2\pi i(ux \cos\theta + uy \sin\theta)\} \tag{3.9}$$

where $i = \sqrt{-1}$, u is the frequency of the sinusoidal wave, θ controls the orientation of the function, and σ is the standard deviation of the Gaussian envelope. To make it more robust against brightness, a discrete Gabor filter, $G[x, y, \theta, \sigma, s]$, is turned to zero DC (direct current) with the application of the following formula:

$$\tilde{G}[x, y, \theta, u, \sigma] = G[x, y, \theta, u, \sigma] - \frac{\sum_{i=-n}^{n} \sum_{j=-n}^{n} G[i, j, \theta, u, \sigma]}{(2n + 1)^2}, \tag{3.10}$$

where $(2n + 1)^2$ is the size of the filter. In fact, the imaginary part of the Gabor filter automatically has zero DC because of odd symmetry. The adjusted Gabor filter is used to filter the preprocessed images.

The success of 2-D Gabor phase coding depends on the selection of Gabor filter parameters, θ, s, and u. In our system, we applied a tuning process to optimize the selection of these three parameters. As a result, one Gabor filter with optimized parameters, $\theta = \frac{\pi}{4}$, $u = 0.0916$ and $\sigma = 5.6179$, is exploited to generate a feature vector with 2,048 dimensions.

Binarized feature vectors such as those used in palm and iris codes do not behave well for indexing and filtering purposes. We use the response values of the filters to carry out the indexing. Each feature

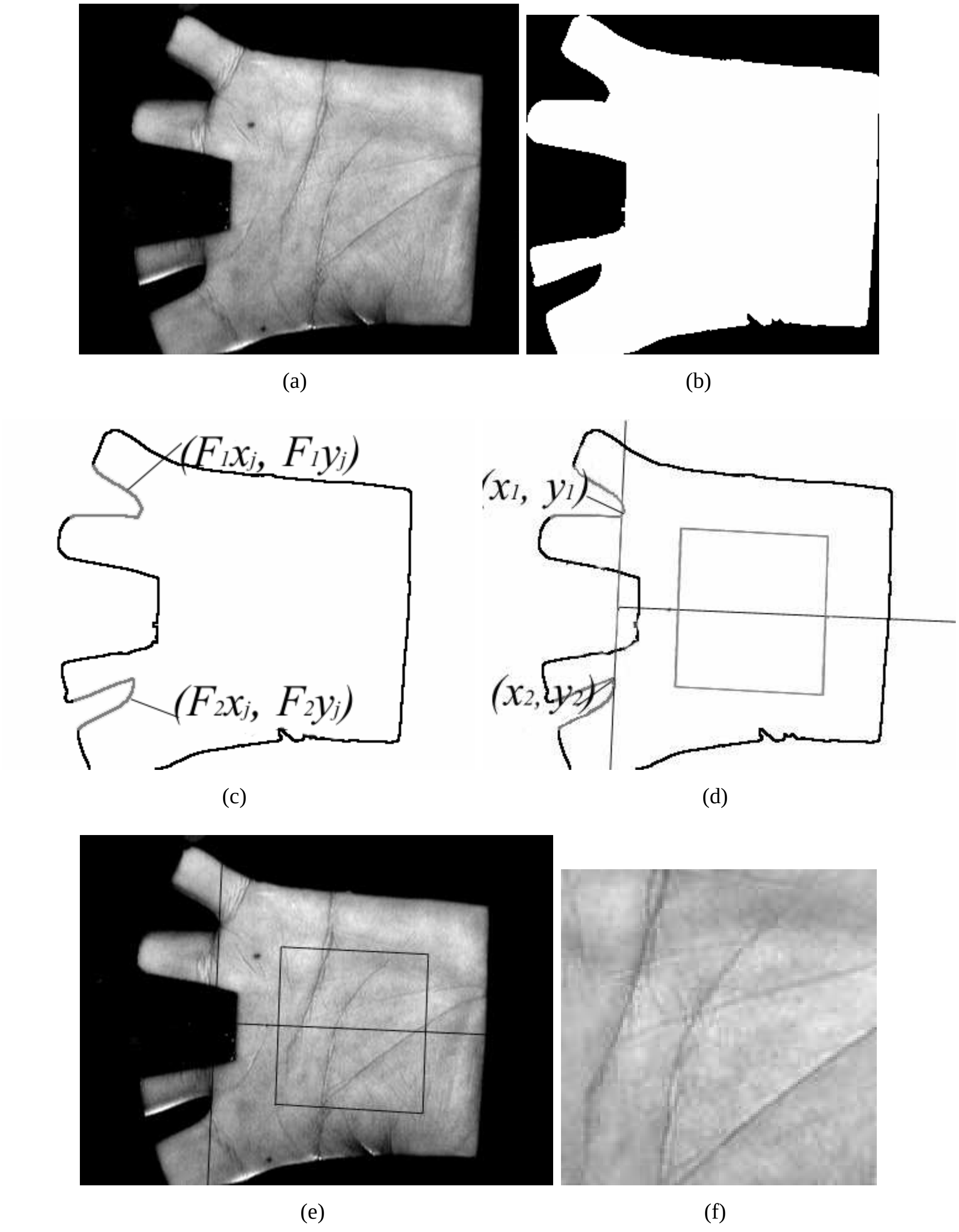
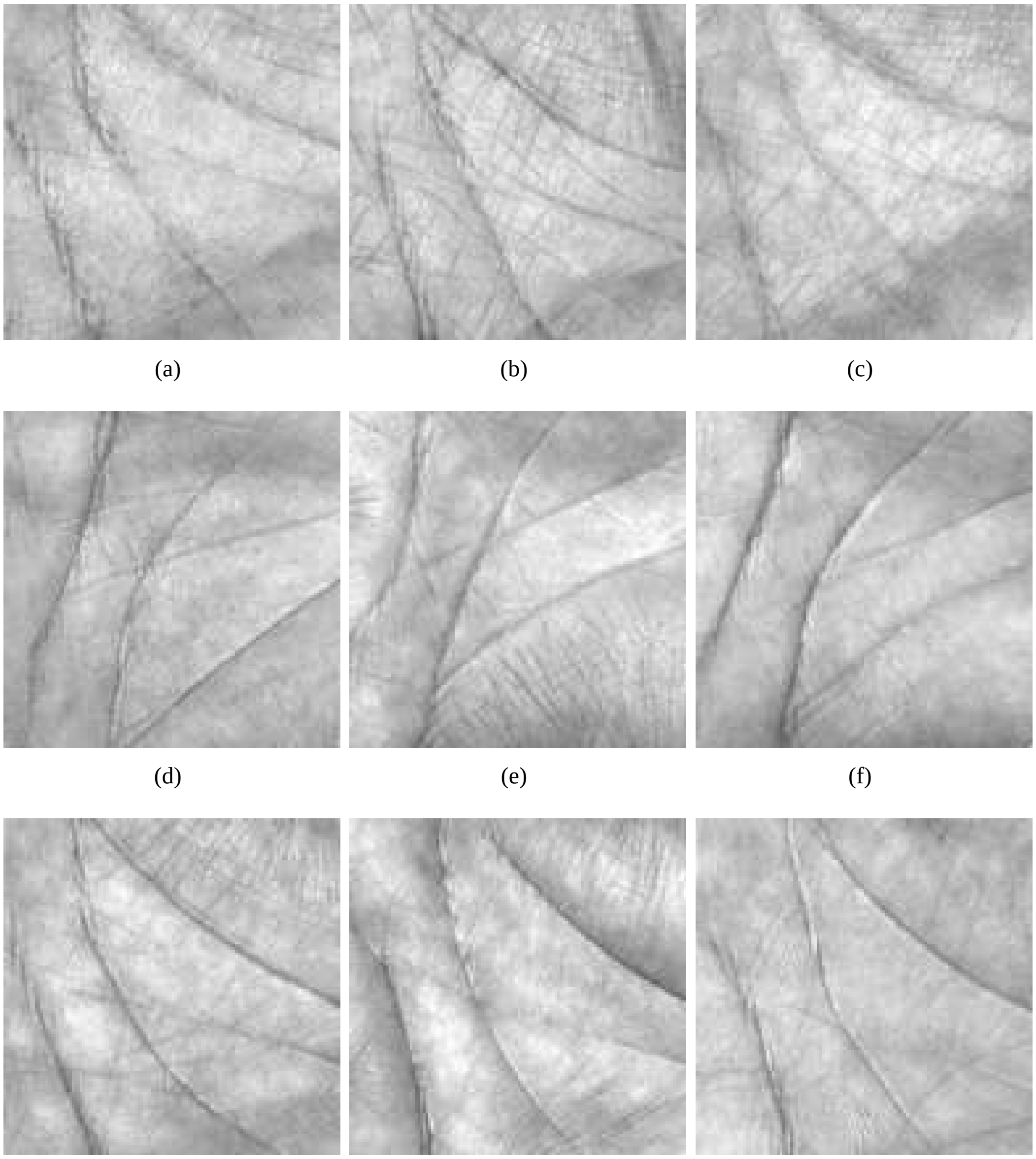


Figure 3.3 Palmprint extraction process. First the image is binarized, then boundary line is extracted and two points are calculated. Then Central region is cropped.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 3.4 Three sets of the palmprint images with the similar principal lines.

Figure 3.5 Preprocessed palmprint images without clear wrinkles.

is first normalized to the range $[-1, 1]$ using the sigmoidal function:

$$y = \frac{1 - e^{-sx}}{1 + e^{-sx}} \tag{3.11}$$

where x is a feature value of the sample and s decides the slope of the sigmoid function. We have selected $s = 1.5$ for palmprint dataset and $s = 10$ for iris. Once all the samples are projected on the random basis, they are scaled to the range $[0, 10]$. Note that the range of projected values depend on the length of the feature vector.

3.3.3 Determining Window Width and Cascade Sequence

As the filtering is a precursor to the regular identification stage, it is desirable to tune this stage in such a way that the accuracy of the identification system is not adversely affected. The width of the window should be selected such that the FNIR is very close to zero. In other words the number of genuine samples outside the window should be practically zero.

Once the window width is finalized, one can re-order the cascade to make the overall process more efficient. As noted before, the order of cascade does not affect the final accuracy. However, we use only a subset of the projections that has very low false rejects. We randomly generate 1500 projections and select best 500 projections based on the scores as mentioned before. The projections are ordered in the sequence of decreasing scores as computed by Equation 3.8. This would minimize the total amount of comparisons as the samples that are rejected in one projection is not considered in the following projections in the cascade.

3.3.4 Effect of Feature Representation

In our experiments, we use three different feature representations for the initial feature vector (before projection) for the purpose of comparison. The first one (referred to as **F1**) is a Gabor wavelet based texture feature that is popular in Iris as well as Palmprint recognition. The response is computed by convolving the image with the following kernel:

$$G(x, y, \theta, u, \sigma) = \frac{1}{2\pi\sigma^2} e^{\{-\frac{x^2+y^2}{2\sigma^2}\}} e^{\{2\pi\iota(uxcos\theta+uysin\theta)\}}, \tag{3.12}$$

where $\iota = \sqrt{-1}$;u is the frequency of the sinusoidal wave; θ controls the orientation of the kernel and σ is the standard deviation of the Gaussian envelope. Palmprint is represented using a 2048 $(32 \times 32 \times 2)$ dimensional feature vector, while the iris was represented using 9600 $(240 \times 20 \times 2)$ dimensional vector. In our case $\theta = \frac{\pi}{4}$ and $\sigma = 0.0916$. Another set of features that was proposed for iris indexing was DCT coefficients in various subbands [53]. After normalizing for pose and illumination variations using an adaptive histogram equalization, image is divided into non-overlapping 8×8 pixel blocks and are transformed to generate DCT coefficients.

The coefficients from each block belonging to a particular subband are grouped together. Energy value E_i of each subband S_i is obtained by summing up the square of coefficients as

$$E_i = \sum S_i(x, y)^2 \tag{3.13}$$

The feature vector consists of different energy values obtained from 10 subbands, resulting in a 10-dimensional feature vector. The image key consists of bin number corresponding to each subband. The bin numbers for each subband are combined together in increasing order of frequency. We refer to the DCT feature representation as **F2** from now on.

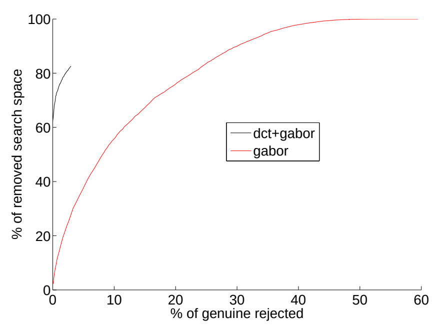


Figure 3.6 Effect of F1 and F3 on filtering performance using only random projections.

Using only the Gabor response features and using a mixture of projections computed from LDA as well as random generation, we can prune 56% of the dataset with the genuine loss of 10%. However, if we use a concatenated feature vector of both Gabor Responses and DCT (referred to as **F3**). In case of palmprint images, we can prune the dataset upto 62.1% with no loss in genuine (see Figure 3.6).

3.3.5 Effect of Window Size

The size of the window selected also plays an important role in determining the accuracy of the system, if the size of the window is too small, we will be able to remove more of the gallery, but many

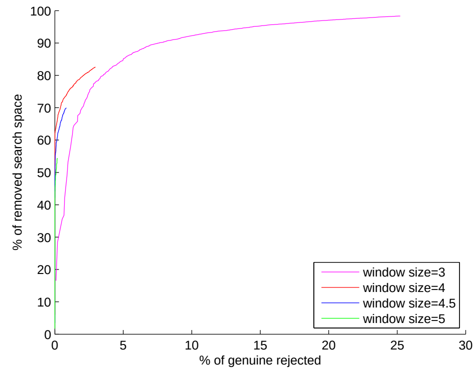


Figure 3.7 Effect of window size on filtering using feature set F3.

genuine samples will also be removed during the process. However, if the window size is too large, it will not be able to reduce the search space considerably (see Figure 3.7). We need to select the optimum window size depending on the nature of the data.

3.4 Experimental Results and Analysis

The experiments are performed on two different modalities: Palmprints and Iris. The PolyU database [2] of palmprint and CASIA database [1] of Iris were used for this purpose. For the palmprint, only those images in the dataset, where the complete palm was visible was considered. For iris, the entire database was considered irrespective of whether part of the iris was covered by the eyelids or not. The Palmprint dataset contains images of size 384×284 with 10 samples from each of 385 users. The iris database contains images of size 320×280 with 3 samples each from a total of 286 users. Tables 3.1 and 3.2 provides the Penetration rates achieved versus the FNIR for different databases, features and projection methods.

Feature	Palmprint PolyU		Iris CASIA	
	Pen.	FNIR	Pen.	FNIR
F1	64.8%	10.3%	38.9%	21.2%
Pyramid Indexing	98.9%	0.15%	88.8%	5.4%
F2	-	-	35.6%	2.6%
F3	37.9%	0%	33.3%	10.0%

Table 3.1 FNIR and filtering rates with various features on Palmprint and Iris datasets.

Feat.	Random		LDA		Combined	
	Pen.	FNIR	Pen.	FNIR	Pen.	FNIR
F1	65.0%	10.3%	46.1%	10.0%	42.4%	10.7%
F3	37.0%	0%	41.0%	5.1%	24.8%	1.0%

Table 3.2 FNIR and filtering rates with various methods of selection the projections using the palmprint dataset.

3.4.1 Cost analysis

We now analyze the cost advantage of carrying out a filtering stage before explicit matching. Each stage of filtering would remove a part of the dataset from consideration, thereby improving the speed of the overall system. However, as the number of projections in the cascade increases, the returns starts diminishing, and at some point the cost of the projection and matching would override the cost advantage due to filtering. Figure 3.8 shows a plot of the reduction in search space ($100-$ penetration) versus the FNIR for various lengths of the cascade. The green dots indicate the lengths of $1, 51, 101, \ldots, 451$. We note that after around 100 projections, the reduction in penetration rate is not significant.

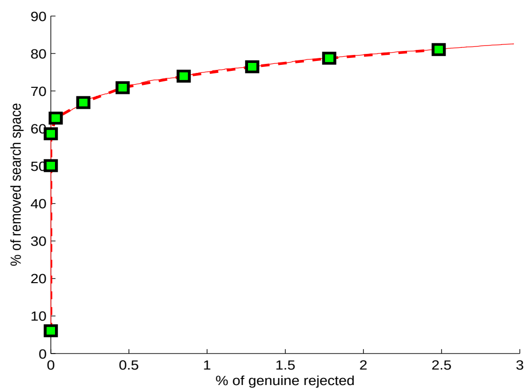


Figure 3.8 Data pruned after each set of 50 projections, starting with 1. The improvement in pruning reduces as the number of projections increase.

To compute the actual gain in speed, we carry out an experiment with a sample probe and the time taken for identification for various lengths of the cascade was determined. Figure 3.9 shows a graph between the overall time taken for identification (in seconds) and the number of projections in the cascade. This experiment was conducted on the PolyU Palmprint database. As expected, the returns of adding further filtering stages reduces and then reverses as the number of projections cross a limit (104). The time required for explicit comparison of a template against all samples in the database was around 2.86 seconds. However, as part of the samples are filtered out, the total time required for comparison decreases, and with a filtering pipeline of 104 random projections, the time required for an identification drops to 0.84 seconds. Note that the actual time will depend on the specific probe being used. However, the overall trend remains the same.

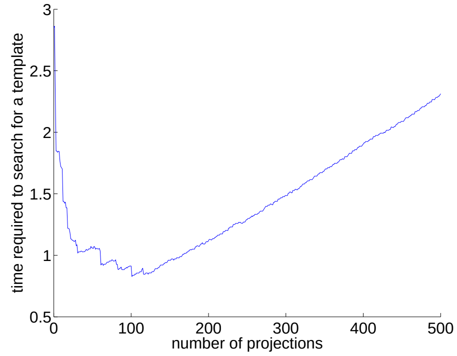


Figure 3.9 Overall time taken (seconds) for identification as the number of projections in the cascade increases.

If the size of dataset is very large such that it cant be fit in memory, then it is divided into chunks. Each chunk is pruned independently. As the process directly lends to parallelization, each chunk can be pruned at the same time on a different machine.

3.5 Hard Biometrics and Soft Biometrics

Soft biometrics have been recently proposed for improving the verification performance of biometric recognition systems. Examples of soft biometrics are skin, eyes, hair colour, height,and ethnicity. Some of them are often cheaper than ”hard”, standard biometrics (e.g., fingerprint, face, iris, palm vein) [59] to extract. Soft biometrics exhibit a low discriminant power for recognizing persons, but can add some evidences about the personal identity, and can be useful for disambiguating certain users.

A second classification based on the disciminability is that of weak vs. strong biometric. Both are sufficient to identify people in most usage scenarios, but stronger biometrics tend to give higher accuracy of lower false match rates. The experimental result performed here are on principal lines of palmprints and iris texture, where the first one is a weak biometric, while the second one is strong.

The results indicate that indexing of weaker biometrics is an easier task compared to strong biometrics, as indicated by the poor indexability on iris. In case of fingerprints conventional minutiae representation is a variable length representation and is not suited for linear projections. We will explore the issue of representation for fingerprint indexing in detail in the next chapter.

3.6 Summary

In this chapter, we have shown the application of random projection on the cascaded filtering on palmprints and iris datasets. The existing works on palmprints are limited to classification based indexing. Random projections was used in the past for dimensionality reduction. Cascaded filtering has been used for classification and detection problems. We use a combination of the two to automatically design a filtering pipeline with high accuracy and low time complexity.

Chapter 4

Cascaded Filtering for Biometric Identification using Random Projections

4.1 Introduction

Among the biometric traits, fingerprints are the most widely studied and accepted form in identification systems. Most biometric systems that scale to national population uses fingerprints as the one of the modalities for identification due to the ease of acquisition, amount of discriminative information available in fingerprints [61], acceptability in legal situations, as well as the availability of low cost devices for authentication purposes [11].

Matching of two fingerprint images is a computationally demanding task due to non-linear deformation of the skin during the acquisition process. The problem is compounded for identification tasks in large databases. To reduce the amount of matching to be performed, a common approach that is employed is the classification of the fingerprints into a set of basic classes [46, 42, 14]. All fingerprints in the database are classified into one of the basic classes (loops, whorl, arches) and stored in partially overlapping partitions. The input fingerprint is also classified, and is only compared against the fingerprints of the corresponding class in the partial database. If fingerprints were equally distributed into say five classes, the penetration rate would be reduced to $P = 0.2$. Therefore, the processing time and the False Identification Rate (FIR) would be reduced. However, as the number of classes is small and the fingerprints are unequally distributed among them (more than 90% of the fingerprints are either right loops, left loops or whorl [41]) the penetration is usually larger. Furthermore, the classification error and rejected fingerprints must be considered when classification is performed automatically. These factors reduce the effectiveness of classification based approach to narrow down the search space.

Fingerprint indexing algorithms reduce the number of comparison by selecting the most probable candidates and sorting them by the similarity to the input [17]. As indexing techniques perform better than exclusive classification considering the size of space that need to be searched [6], many indexing algorithms have been proposed recently. Germain *et al.* proposed a flash algorithm for fingerprint indexing [27]. Bebis *et al.* [4] proposed the Delaunay triangulation of minutia points to perform fingerprint indexing. Boer *et al.* [10] used the registered directional field estimate, FingerCode and minutiae triplet along with their combination to index fingerprint databases. Bhanu and Tan [6] generated minutiae triplets and used angles, handedness, type, direction, and maximum side as the features for indexing. They also applied some constraints on minutiae selection to avoid spurious minutiae. Jain *et al.* [39] use the features around a core point of a Gabor filtered image to realize indexing. Another indexing algorithm was proposed based on correlation of the robustly detected singular points in [49].

Most of the indexing methods available for fingerprints are based on the detection of core and delta points, referred to as the singular points. The accuracy of the entire system is dependent on the accuracy of detection of singular points. Other indexing schemes rely on alignment of the fingerprints for a compact representation, and the indexing accuracy is often dependent on the quality of alignment.

4.2 Feature Extraction

One of the major problems with fingerprint identification is that the feature vector is variable in length. Different samples of fingerprints from the same user can have different number of minutiae extracted. This prohibits us from using any indexing method that assumes that the pattern is a point in a Euclidean feature space. Another problem faced in fingerprint identification is the lack of alignment of the query with samples in the database.

To overcome the issue with minutiae representation, various fixed length feature representations have been proposed such as low order Delaunay triangulation [4], minutiae triplets [6], and Finger Code [39]. In this work, we start with the assumption that a set of fixed length features are available for representing each fingerprint sample, and derive a method for efficient filtering using the given set of features.

We have currently used two different sets of features and they are concatenated together so that every finger is represented using a fixed length feature vector. The first set of features are extracted from the triangles formed by minutiae in the image (referred to as minutiae triplets), and the second set of features are extracted from the quadrilaterals formed from the geometrical locations of minutiae, as proposed in [35].

To extract the features from triangles, the largest side, l (see Figure 4.1), and the two angles (α_1 and α_2) that involve l are computed. The feature representation of each triangle is (l, α_1, α_2), where ($\alpha_1 < \alpha_2$).

The features from minutiae quadruplets is as proposed by Iloanusi *et al.* [35], which involves 7 features $< \varphi_1, \varphi_2, \delta_1, \delta_2, \rho_1, \rho_2, \eta >$ (see Figure 4.2). The two features φ_1 and φ_2 are the differences of two opposite angles in the quadrilateral, and δ_1,δ_2 are the lengths of the two diagonals. ρ_1 and ρ_2 are the heights of the parallelogram.

The last feature η is a global feature, and is a combination of sides and area of the quadrilaterals.

$$\eta = 100 \log_{10}(\tau\nu), \tag{4.1}$$

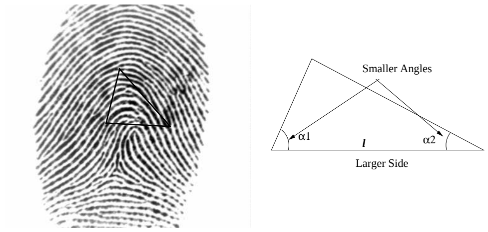


Figure 4.1 Triangles formed from the minutiae and the extracted features.

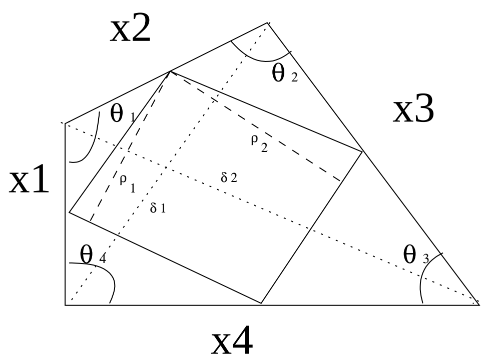


Figure 4.2 A minutia quadruplets

where

$$\tau = \sqrt{A_p} + \sqrt[4]{x_1 \times x_2 \times x_3 \times x_4} \tag{4.2}$$

$$\nu = \sqrt{A_q} + \sqrt{y_1 \times y_2} \tag{4.3}$$

A_p is the area of the parallelogram, x_1, x_2, x_3 and x_4 are the lengths of the sides of quadrilateral. A_q is the area of the quadrilateral, and y_1 and y_2 are the length of the sides of the parallelogram. We have removed concave quadrilaterals and all quadrilaterals with crossed edges were uncrossed to form regular convex quadrilaterals. See Figure

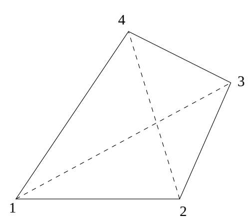


Figure 4.3 A convex quadrilateral

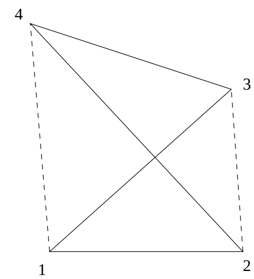


Figure 4.4 An inverted quadrilateral

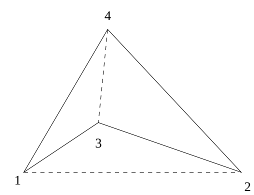


Figure 4.5 A concave quadrilateral

The above procedure gives us the features for each triangle and quadrilateral. The number of triangles and quadrilaterals vary considerably between images as it depends on the number of minutiae that are detected. In our experiment, we fixed the number of triangles and quadrilaterals that were selected, which were empirically set to be 800 and 1200 respectively. To reduce the influence of deformations in fingerprints, we concentrate on local minutiae structure, and hence only the smaller triangles and quadrilaterals are considered in computing the feature vectors. In short, the smallest 800 and 1200 triangles and quadrilaterals were used in construction of the final feature vector.

The final feature vector contains the frequency count of different triangles and quadrilaterals present in the fingerprint. To determine this, we first learn the most prominent k clusters are determined from the training samples for both triangles and quadrilaterals using $k - means$ clustering in the corresponding feature spaces. Let c_k denote the k^{th} cluster.

To extract the feature from images, its triplets $\{t_w | w = 1, 2, .., Q\}$ are assigned to the nearest cluster based on the Euclidean distance to the cluster centers.

$$\text{Assign } t_w \text{ to } c_k, \text{ if } k = argmin\{|t_w - n_j|, j = 1..k\} \tag{4.4}$$

Here, n_j is the centroid of the j^{th} cluster and c_k is the cluster id. Each triplet is assigned to a single cluster. The feature vector is constructed by counting the number of triplets assigned to each cluster. Thus, the feature vector of image Y is $F_t(Y) = \{a_1^Y, a_2^Y, \ldots, a_k^Y\}$, where a_i^Y is the number of triplets from image Y that are assigned to cluster i and k is the total number of clusters. We will refer to a_i as an accumulator for centroid i. The quadruplets based features are also converted to a histogram using the accumulation process as above.

In our experiments, the numbers of clusters k was empirically set to 50 for both triplets and quadruplets, resulting in feature vectors of length 50 for each. The two feature vectors are then concatenated to form a single 100 dimensional feature vector. The resulting feature vector may be written as the concatenation: $F(Y) = [F_t(Y)\ F_q(Y)]$. As the individual features are rotation invariant, there is no need of alignment between samples for the purpose of matching.

4.3 Experimental Results and Analysis

We used the FVC2002 (DB1,DB2,DB3,DB4) [51] for evaluating the proposed algorithm. Each DB consists of eight prints each of 100 distinct fingers captured by optical sensors (500 dpi). We separate each dataset into independent training and testing sets of equal size.

First the window size W_j for each projection is calculated from the training samples. The size of the window should be such that the samples of the same class falling outside the should be minimal and number of sample falling within the window should be maximum. For each probe template (testing sample), the features are extracted and projected into a lines. We will search only in the dataset which falls in the window centered around the probe. To extract the features from fingerprints we have used 50 clusters for both quadrilaterals and triangles, giving us a combined feature vector of size 100. The number of projections to be used for cascading and size of the window is decided by the experiments and the dataset we are using.

To generate the cascade, we start with a set of 3000 random projections and select the best 100 projections. The final cascade is typically terminated within 50 projections. We also added 100 LDA and 100 PCA projections values for comparison purposes. However, there were no perceptible changes in the results, confirming the effectiveness of random projections for the problem.

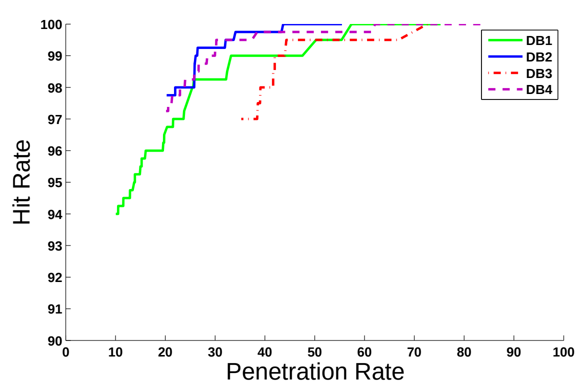


Figure 4.6 Hit Rate vs. Penetration Rate on FVC 2002 DB1,..,DB4.

Figure 4.6 shows the hit rate vs penetration rate on different DBs of FVC 2002. Figure 4.7 clearly shows our method is significantly better in terms of efficiency as compared to [35], and the difference increases with the size of database. Figure 4.8 shows the nature of filtering rate with increasing number of projections. We note that significant portion of the filtering takes place within the first 10 projections in the cascade, and one might stop there for efficiency purposes. We note that as the number of training samples increase, we can do a better estimation of the window size and the performance improves as seen in Figure 4.9.

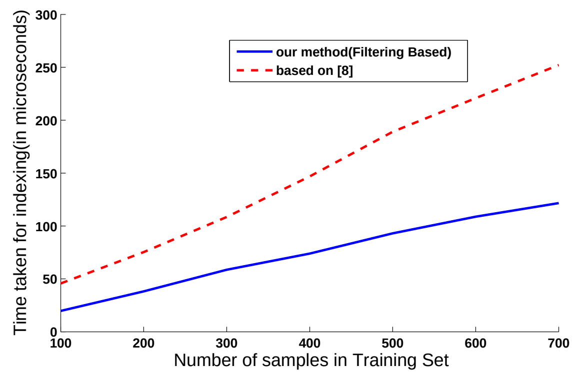


Figure 4.7 Time taken for indexing with increasing number of training samples.

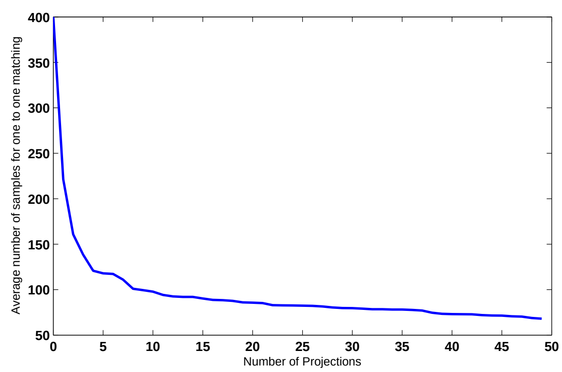


Figure 4.8 Decrease in penetration with increasing number of projections.

One of the advantages of the proposed method is that it efficiently and effectively combine different feature sets. Figure 4.10 shows the results of using the two feature sets considered independently and when combining the two. As the final cascade is based on projections, the length of the feature vector does affect the database and the effect on the filtering process is minimal.

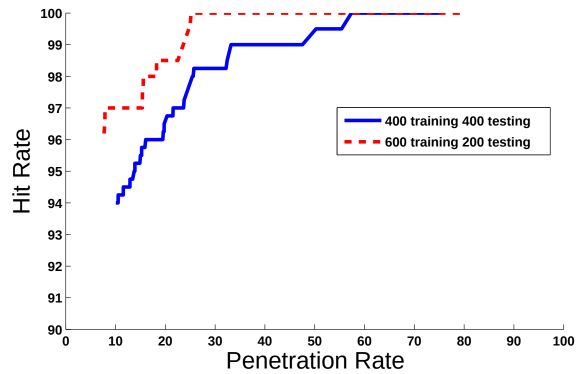


Figure 4.9 Effect of training sample size on filtering performance.

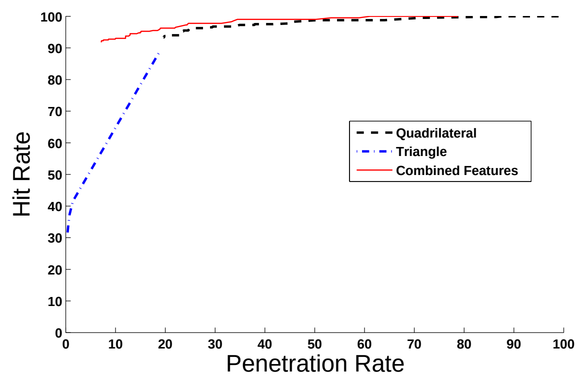


Figure 4.10 Effect of using the individual (triangles and quadrilaterals) and combined feature sets.

Another common indexing approach is KD-Tree. A kD tree (kd-tree, or k-Dimension tree) is a data structure that is used for storing coordinates so nearest-neighbour searches or range-searches are quick. It is a binary-search tree that is specialised for coordinate searching, and is useful for answering questions such as which point is closest to a query data. We have seen the usage of KD-Tree in [43] in the field of biometrics. However, the technique achieves acceptable penetration-hit tradeoff with the use of multiple biometric traits.

Figure 4.11 shows the comparison of the proposed approach to the kd tree based indexing. The result shows the kd tree doesnt perform very good for fingerprints.

Method	Penetration at 99% Hit Rate	Time taken in μ $secs.$
Quadruplets[35]	**20%**	147
Combined Features	26%	**74**

Table 4.1 Result in FVC 2002 DB2 datasets, with equal number of training and testing samples.

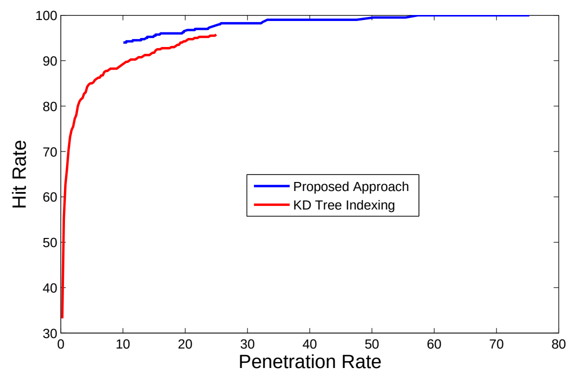


Figure 4.11 Comparison of Hit Rate vs. Penetration Rate with proposed approach and kd tree

4.4 Summary

In this chapter we have extended the cascaded filtering approach to fingerprints datasets. The features extracted from the minutiae quadruplets and triplets are combined for projection. We have seen that the result we get is comparable to [35], while using less than half the time. We have seen that adding more features increases the accuracy. The results show that the proposed approach is effective in a variety of application scenarios, provided we use an appropriate representation.

Chapter 5

Conclusions and Future Work

This thesis presents a novel approach towards cascaded filtering for biometric identification using random projections. Biometrics has become important field of research over the past few years with increased deployment of biometric systems in commercial applications. With the governments all over the world trying to keep the record of their citizens, and visitors crossing the border, the speed of search for a record in the database becomes critical. Increase in security concerns has also pushed the importance of biometrics in day to day life.

Unfortunately, biometric data does not lend itself to indexing process as the the data is not evenly spread in the feature space. In this work, we have proposed a generic method for cascaded filtering using projections and have shown its application to different biometric traits. With filtering, we reduce the amount of time taken to retrieve the candidate list for searching without compromising in accuracy. The results show that we can reduce the search space by over 63% with no increase in the false non-identification rate (FNIR). The approach is flexible to use different feature sets and their combinations to carry out the projection. As each sample can be projected independently during the training, the computational cost of inserting a new sample into the database is minimal. The approach also allows a high degree of parallelization or pipelined processing.

Random projections are a powerful method of dimensionality reduction that provide us with both conceptual simplicity, and very strong error guarantees. The simplicity of projections allows them to be analyzed thoroughly, and this, combined with the error guarantees, makes them a very popular method of dimensionality reduction. In this thesis we used random projections for filtering of biometrics data, which is initially applied to palmprint datasets. Most existing methods for indexing, including palmprints, are based on hand-crafted features. In contrast the proposed method allows us to create automatic filtering pipelines from a given set of features.

We also explored a method for fingerprint database filtering using cascaded projections. The results are comparable to the state of the art fingerprint indexing methods. Experiments show that the proposed method extremely efficient and can give a significant advantage when used as the first stage in identification. While we do not propose any new features for fingerprint indexing, our method is able to combine a large number of existing feature descriptors into a compact and efficient cascaded filter, irrespective of the feature vector size. This results in significant savings in time during identification of fingerprints. Due to it efficiency, the method may be used as the first stage while combining multiple indexing and filtering methods.

The method is scalable as well as incremental. We provide a promising step towards developing a general indexing framework. The ability of our approach to be parallelized makes it useful in indexing with computer clusters. The process is simple and avoids retraining if we want to introduce new samples to the database, without affecting the overall feature space distribution. Comparison of the results with other state of the art approaches show that while providing lower penetration rate, they take almost double time for a search compared to the proposed approach.

The use of random projections also avoids computationally and memory intensive training for finding popular projections such as PCA and LDA. This allows us to deal with very large datasets.

One of the fundamental aspects that need further exploration is the nature of features that lend itself suitable or unsuitable for indexing as observed in the case of palmprints vs. iris. Ironically the very nature of strong biometric features that introduce high (and uniform) inter-class variability makes them unsuitable for indexing. One can also look at formulation of the projection criterion function that lends itself to analytical optimization, leading to non-random projection cascades. The method of cascaded filtering using random projections can also be applied to other fields such as object detection, recognition, etc.

Related Publications

- Atif Iqbal and Anoop M. Namboodiri, “Cascaded Filtering for Biometric Identification using Random Projection”, in Proceedings of the IEEE National Conference on Communications (NCC), 2011, pp. 1-5
- Atif Iqbal and Anoop M. Namboodiri, “Cascaded Filtering for Fingerprint Identification using Random Projection”, in Proceedings of the IEEE Compter Society Conference Computer Vision and Pattern Recognition Workshops (CVPRW) 2012

Bibliography

[1] Center for biometrics and security research.

[2] Polyu palmprint palmprint database.

[3] Small codes and large image databases for recognition. *Proc. of Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[4] G. Bebis, T. Deaconu, and M. Georgiopoulos. Fingerprint identification using delaunay triangulation. *IEEE International Conference on Intelligence, Information and Systems*, pages 452–459, 1999.

[5] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 509–522, 2001.

[6] B. Bhanu and X. Tan. Fingerprint indexing based on novel features of minutiae triplet. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 25:616–622, 2003.

[7] E. Bingham and H. Mannila. Random projection in dimensionality reduction: applications to image and text data. *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 245–250, 2001.

[8] W. W. Bledsoe. In Memorium Woodrow Wilson Bledsoe. *http://www.cs.utexas.edu/users/boyer/bledsoe-memorial-resolution.pdf*.

[9] B. Boeckmann, A. Bairoch, R. Apweiler, M. C. Blatter, A. Estreicher, E. Gasteiger, M. J.Martin, K.Michoud, I. Phan, R. Pilbout, and M. Schneider. The swiss-prot protein knowledgebase and its supplement trembl. *Nucleic Acids Research*, 31:365–370, 2003.

[10] J. D. Boer, A. M. Bazen, and S. H. Gerez. Indexing fingerprint databases based on multiple features. *ProRISC Workshop on Circuits, Systems and Signal Processing*, pages 300–306, 2001.

[11] R. Bolle and S. Pankanti. *Biometrics, Personal Identification in Networked Society: Personal Identification in Networked Society*. Kluwer Academic Publishers, Norwell, MA, USA, 1998.

[12] R. Cappelli, M. Ferrara, and D. Maltoni. Minutia cylinder-code: a new representation and matching technique for fingerprint recognition. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 32(5):2128–2141, 2010.

[13] R. Cappelli, M. Ferrara, and D. Maltoni. Fingerprint indexing based on minutia cylinder-code. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 33(5):1051–1057, 2011.

[14] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni. Fingerprint classification by directional image partitioning. *IEEE Trans. Pattern Analysis Machince Intelligence*, 5(21):402–421, 1996.

[15] H. Chen and R. Rasmussen. Intellectual access to images. *Library Trends*, 48(2):291–302, 1999.

[16] J.-Y. Chen, C. A. Bouman, and J. C. Dalton. Hierarchical browsing and search of large image databases. *IEEE Transactions on Image Processing*, 9(3):442–455, 2000.

[17] K. Choi, D. Lee, S. Lee, and J. Kim. An improved fingerprint indexing algorithm based on the triplet approach. *AVBPA*, pages 584–591, 2003.

[18] A. Das and A. Jain. *Indexing the World Wide Web: The Journey So Far*. 2011.

[19] S. Dasgupta. Experiments with random projection. *Uncertainty in Artificial Intelligence: Proceedings of the Sixteenth Conference*, pages 143–151, 2000.

[20] J. Daugman. How iris recognition works? *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):2130, 2004.

[21] S. Davies. Touching big brother: How biometric technology will fuse flesh and machine. *Information Technology and People*, 7(4), 1994.

[22] M. de Kunder. http://www.worldwidewebsize.com. May 2012.

[23] L. Devroye and T. J. Wagner. Nearest neighbour methods in descrimination. *Handbook of Statistics*, 2, 1982.

[24] L. Fang, M. K. H. Leung, T. Shikhare, V. Chan, and K. F. Choon. Palmprint classification. *International Conference on Systems, Man and Cybernetics*, pages 2965–2969, 2006.

[25] X. Z. Fern and C. E. Brodley. Random projection for high dimensional data clustering: A cluster ensemble approach. *International Conference on Machine Learning*, pages 186–193, 2003.

[26] S. F. Galton. Fingerprints. 1892.

[27] R. Germain, A. Califano, and S. Colville. Fingerprint matching using transformation parameter clustering. *IEEE Computational Science and Engineering*, 4:42–49, 1997.

[28] A. Gersho and R. M. Gray. Vector quantisation and data compression. *Kluwer*, 1991.

[29] N. Goel, G. Bebis, and A. Nefian. Face recognition experiments with random projection. *Proceedings of SPIE*, 5779(1):426–437, 2005.

[30] I. B. Group. Independent testing of iris recognition technology: Final report. *Available at http://www.biometricgroup.com/reports/public/ITIRT.html*, 2005.

[31] F. Hao, J. Daugman, and P. Zielin'ski. A fast search algorithm for a large fuzzy database. *IEEE Transactions of Information Forensics and Security*, 3(2):203–212, 2008.

[32] C. Hegde, M. A. Davenport, M. B. Wakin, and R. G. Baraniuk. Efficient machine learning using random projections. Dec. 2007.

[33] E. R. Henry. Classification and uses of fingerprints. *Routledge: London*, 1900.

[34] IAFIS. Integrated automatic fingerprint identification system. *http://www.fbi.gov/hq/cjisd/iafis.htm*.

[35] O. Iloanusi, A. Gyaourova, and A. Ross. Indexing fingerprints using minutiae quadruplets. *IEEE Compter Society Conference Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 127–133, 2011.

[36] P. Indyk and R. Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. *Proceedings of 30^{th} ACM Symposium on Theory of Computing*, pages 604–613, 1998.

[37] D. J. High confidence visual recognition of persons by a test of statistical independence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1148–1161, 1993.

[38] A. Jain and S. Pankanti. *Fingerprint Classification and Matching*. Academic Press, 2000.

[39] A. Jain, S. Prabhakar, L. Hong, and S. Pankanti. Fingercode: a filterbank for fingerprint representation and matching. *IEEE Computer Vision and Pattern Recognition*, 2:187–193, 1999.

[40] A. K. Jain and S. Z. Li. Handbook of face recognition. 2005.

[41] A. K. Jain and D. Maltoni. *Handbook of Fingerprint Recognition*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2003.

[42] A. K. Jain, S. Prabhakar, and H. Lin. Multichannel approach to fingerprint classification. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 21(4):348–359, 1999.

[43] U. Jayaraman, S. Prakash, and P. Gupta. An efficient technique for indexing multimodal biometric databases. *Int. J. Biometrics*, 1(4):418–441, jul 2009.

[44] A. T. B. Jin. Cancellable biometrics and multispace random projections. *Conference on Computer Vision and Pattern Recognition Workshop(CVPRW'06)*, June 2006.

[45] W. B. Johnson and J. Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. *Contemporary Mathematics*, 26:189–206, 1984.

[46] K. Karu and A. K. Jain. Fingerprint classification. *Pattern Recognition*, 33(29):389–404, 1996.

[47] J. Kleinberg. Two algorithms for nearest-neighbor search in higher dimensions. *Proc. of ACM Symposium on Theory of Computing*, pages 599–608, May 1997.

[48] T. Liu, C. Zhang, and P. Hao. Fingerprint indexing based on las registration. *IEEE International Conferenece on Image Processing*, pages 301–304, 2006.

[49] T. Liu, G. Zhu, C. Zhang, and P. Hao. Fingerprint indexing based on singular point correlation. *IEEE International Conferenece on Image Processing*, pages 293–296, 2005.

[50] A. Magen. Dimensionality reductions that preserve volumes and distance to ane spaces. *NEC Research Institute*, 2002.

[51] D. Maio, D. Maltoni, R. J. Cappelli, L. Wayman, and A. K. Jain. Fvc2002: Second fingerprint verification competition. *Proc. International Conference on Pattern Recognition*, pages 811–814, August 2002.

[52] B. McGuigan. http://www.wisegeek.com/how-big-is-the-internet.htm. May 2012.

[53] H. Mehrotra, B. G. Srinivas, B. Majhi, and P. Gupta. Indexing iris biometric database using energy histogram of dct subbands. *Communications in Computer and Information Science*, 40:194–204, 2009.

[54] A. Mhatre, S. Chikkerur, and V. Govindaraju. Indexing biometric databases using pyramid technique. pages 841–849, 2005.

[55] A. Mhatre, S. Chikkerur, and V. Govindaraju. Indexing biometric databases using pyramid technique. *Proceedings of Audio and Video-based Biometric Person Authentication*, pages 841–849, july 2005.

[56] R. Micheals, P. Grother, P. Grother, R. J. Micheals, P. J. Phillips, and P. J. Phillips. Face recognition vendor test 2002 performance metrics. *Proceedings 4th International Conference on Audio Visual Based Person Authentication*, pages 937–945, 2003.

[57] A. Moenssens. *Fingerprint Techniques*. Chilton Book Company, London, 1971.

[58] F. Monrose and A. Rubin. Authentication via keystroke dynamics. *Proceedings of Fourth ACM Conference on Computer and Communications Security*, pages 48–56, 2005.

[59] K. Niinuma, U. Park, and A. K. Jain. Soft biometric traits for continuous user authentication. *Transactions Informatics Forensics Security*, 5(4):771–780, december 2010.

[60] L. OGorman. Seven issues with human authentication technologies. *Proc. of Workshop on Automatic Identification Advanced Technologies (AutoID)*, pages 185–186, 2002.

[61] S. Pankanti, S. Prabhakar, and A. K. Jain. On the individuality of fingerprints. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24:1010–1025, 2002.

[62] C. Papadimitriou, P. Raghvan, H. Tamaki, and S. Vempala. Latent semantic analysis: A probabilistic analysis. *Proceedings of ACM Symposium On the principles of Database Systems*, pages 159–168, june 1998.

[63] N. B. Puhan and N. Sudha. A novel iris database indexing method using the iris color. *3rd IEEE Conferece on Industrial Electronics and Applications*, 2008.

[64] D. M. R. Cappelli, D. Maio and L. Nanni. A two-stage fingerprint classification system. *Proceedings of ACM SIGMM workshop on Biometrics methods and applications*, pages 95–99, 2003.

[65] N. Ratha, K. Karu, S. Chen, and A. Jain. A real time matching system for large fingerprint databases. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8), August 1996.

[66] A. Ross and R. Mukherjee. Augmenting ridge curves with minutiae triplets for fingerprint indexing. *SPIE Conference on Biometric Technology for Human Identification*, 2007.

[67] A. Ross and R. Mukherjee. Indexing iris images. *IEEE International Conference on Pattern Recognition*, pages 1–4, 2008.

[68] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter-sensitive hashing. *Proceedings of International Conference on Computer Vision*, pages 750–757, 2003.

[69] N. Subcommittee. Biometric history. August 2006.

[70] V. Sulic, J. Pers, M. Kristan, and S. Kovacic. Efficient dimensionality reduction using random projection. *Computer Vision Winter Workshop*, 2010.

[71] Z. Sun, Y. Wang, T. Tan, and J. Cui. Improving iris recognition accuracy via cascaded classifiers. *IEEE Transactions on Systems, Man and Cybernetics*, 35(3):435–441, august 2005.

[72] F. Torpay. Unpublished 1995 report by frank torpay of mitre corporation using data extracted from the fbis identification division automated services database of 22,000,000 human-classified fingerprints.

[73] M. A. Turk and A. P.Pentland. Face recognition using eigen faces. *Computer Vision and Pattern Recongnition*, pages 586–591.

[74] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *Proceedings of Computer Vision and Pattern Recognition*, pages 511–518, 2001.

[75] J. L. Wayman, A. K. Jain, D. Maltoni, and D. Maio. *Biometric Systems: Technology, Design and Performance Evaluation*. Springer Publishing Company, Incorporated, 1st edition, 2010.

[76] C. Wilson, A. R. Hicklin, M. Bone, H. Korves, P. Grother, B. Ulery, R. Micheals, M. Zoepfl, S. Otto, and C. Watson. Fingerprint vendor technology evaluation 2003: Summary of results and analysis report. *NIST Technical Report NISTIR 7123, National Institute of Standards and Technology*, 2004.

[77] P. Wu and M. Li. Pyramid edge detection based on stack filter. *Pattern Recognition Letter*, 18(4):239–248, 1997.

[78] W. Xiangqian, D. Zhang, W. Kuanquan, and B. H. 0003. Palmprint classification using principal lines. *Pattern Recognition*, 37(10):1987–1998, 2004.

[79] X. Yang, J. Feng, and J. Zhou. Palmprint indexing based on ridge features. *International Joint Conference on Biometrics*, 2011.

[80] D. Zhang, W.-K. Kon, J. You, and M. Wong. Online palmprint identification. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1041–1050, sep 2003.