# Multibiometric Systems: Fusion Strategies and Template Security

By

Karthik Nandakumar

A Dissertation

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

Department of Computer Science and Engineering

2008

# Report Documentation Page

*Form Approved*
*OMB No. 0704-0188*



| 1. REPORT DATE | 2. REPORT TYPE | 3. DATES COVERED |
|---|---|---|
| **2008** | | **00-00-2008 to 00-00-2008** |

| 4. TITLE AND SUBTITLE | 5a. CONTRACT NUMBER |
|---|---|
| **Multibiometric Systems: Fusion Strategies and Template Security** | 5b. GRANT NUMBER |
| | 5c. PROGRAM ELEMENT NUMBER |
| 6. AUTHOR(S) | 5d. PROJECT NUMBER |
| | 5e. TASK NUMBER |
| | 5f. WORK UNIT NUMBER |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Michigan State University,Department of Computer Science & Engineering,3115 Engineering Building,East Lansing,MI,48824** | 8. PERFORMING ORGANIZATION REPORT NUMBER |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) | 10. SPONSOR/MONITOR'S ACRONYM(S) |
| | 11. SPONSOR/MONITOR'S REPORT NUMBER(S) |

12. DISTRIBUTION/AVAILABILITY STATEMENT
**Approved for public release; distribution unlimited**

13. SUPPLEMENTARY NOTES

14. ABSTRACT
**see report**

15. SUBJECT TERMS

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT **unclassified** | b. ABSTRACT **unclassified** | c. THIS PAGE **unclassified** | **Same as Report (SAR)** | **249** | |

# Abstract

## Multibiometric Systems: Fusion Strategies and Template Security

By

Karthik Nandakumar

Multibiometric systems, which consolidate information from multiple biometric sources, are gaining popularity because they are able to overcome limitations such as non-universality, noisy sensor data, large intra-user variations and susceptibility to spoof attacks that are commonly encountered in unibiometric systems. In this thesis, we address two critical issues in the design of a multibiometric system, namely, fusion methodology and template security.

First, we propose *a fusion methodology based on the Neyman-Pearson theorem* for combination of match scores provided by multiple biometric matchers. The *likelihood ratio (LR) test* used in the Neyman-Pearson theorem directly maximizes the genuine accept rate (GAR) at any desired false accept rate (FAR). The densities of genuine and impostor match scores needed for the LR test are estimated using finite Gaussian mixture models. We also extend the likelihood ratio based fusion scheme to incorporate the quality of the biometric samples. Further, we also show that the LR framework can be used for designing sequential multibiometric systems by constructing a binary decision tree classifier based on the marginal likelihood ratios of the individual matchers. The LR framework achieves consistently high recognition rates

across three different multibiometric databases without the need for any parameter tuning. For instance, on the WVU-Multimodal database, the GAR of the LR fusion rule is 85.3% at a FAR of 0.001%, which is significantly higher than the corresponding GAR of 66.7% provided by the best single modality (iris). The use of image quality information further improves the GAR to 90% at a FAR of 0.001%.

Next, we show that the proposed *likelihood ratio based fusion framework is also applicable to a multibiometric system operating in the identification mode*. We further investigate rank level fusion strategies and propose a hybrid scheme that utilizes both ranks and scores to perform fusion in the identification scenario.

While fusion of multiple biometric sources significantly improves the recognition accuracy, it requires storage of multiple templates for the same user corresponding to the individual biometric sources. Template security is an important issue in biometric systems because unlike passwords, stolen biometric templates cannot be revoked. Hence, we propose *a scheme for securing multibiometric templates as a single entity using the fuzzy vault framework*. We have developed fully automatic implementations of a fingerprint-based fuzzy vault that secures minutiae templates and an iris cryptosystem that secures iriscode templates. We also demonstrate that a multibiometric vault achieves better recognition performance and higher security compared to a unibiometric vault. For example, our multibiometric vault implementation based on fingerprint and iris achieves a GAR of 98.2% at a FAR of less than 0.01% and provides approximately 49 bits of security. The corresponding GAR values of the individual iris and fingerprint vaults are 88% and 78.8%, respectively. When the iris and fingerprint vaults are stored separately, the security of the system is only 41 bits.

To My Grandfather and Grandmother

# Acknowledgments

First and foremost, I would like to thank my grandfather Shri. P.S. Venkatachari and my grandmother Smt. P.S. Vanaja for their prayers and blessings. Without their support and encouragement at crucial periods of my life, it would not have been possible for me to pursue graduate studies and aim for a career in research. Their hard work and positive attitude even at the age of 80, is my main source of inspiration. I am proud to dedicate this thesis and all the good things in my life to them.

I would like to express my sincere gratitude to my advisor Dr. Anil K. Jain for providing me the opportunity to work in the exciting and challenging areas of pattern recognition and biometrics. His constant motivation, support and infectious enthusiasm have guided me towards the successful completion of my graduate studies. My interactions with him have been of immense help in defining my research goals and in identifying ways to achieve them. His encouraging words have often pushed me to put in my best possible efforts. I would also like to thank him for his guidance and help in identifying a suitable career path for me. Above all, the complete belief that he has entrusted upon me has instilled a great sense of confidence and purpose in my mind, which I am sure will stand me in good stead throughout my career.

I also thank my guidance committee members Dr. George Stockman, Dr. Bill Punch, Dr. Sarat Dass and Dr. Arun Ross for their valuable comments and suggestions that have greatly enhanced and shaped this thesis. In particular, I appreciate

Dr. Stockman for the time and effort that he has spent in guiding my research, assisting me in administrative tasks and moulding me professionally. Special thanks also goes to Dr. Arun Ross and Dr. Sarat Dass for the enlightening research discussions that have shown me the right path on many occasions. I would also like to thank Dr. Sharath Pankanti and Dr. Salil Prabhakar for their assistance in the template security project.

The research in this thesis was supported by grants from the National Science Foundation (NSF-ITR grant number CNS-0325640), the Army Research Office (ARO grant number W911NF-06-1-0418) and the Center for Identification Technology Research at West Virginia University. I would like to thank the NSF, ARO and CITeR for their generous financial support.

I would like to thank all the faculty members in the Department of Computer Science and Engineering and the Department of Statistics and Probability at Michigan State University. In particular, I would like to express my thanks to Dr. Abdol Esfahanian and Dr. Eric Torng for their help as CSE graduate directors, Dr. Li Xiao for her encouragement and support, Dr. Jon Sticklen for his support during my tenure as a teaching assistant and his special interest in my research work and career, Dr. Herman Hughes and Dr. Matt Mutka for their guidance during my first year at MSU and Dr. James Stapleton for his encouragement and help as the graduate director of the Statistics department. I would also like to express my appreciation and gratitude to Linda Moore, Debbie Kruch, Cathy Davison, Starr Portice, Norma Teague, Kim Thompson, Cathy Sparks, Sue Watson and Adam Pitcher for their administrative assistance and support. I would also like to express my thanks to

Mr. Dale Setlak, Dr. Kuntal Sengupta, Mr. Dick Jones and Dr. Mike Boshra at Authentec, Inc. and Dr. Srimat T. Chakradhar, Dr. Anand Raghunathan and Dr. Srivaths Ravi at NEC Labs America, Inc. for the internship opportunities. A special word of appreciation for Dr. Chandrasekara Rao Komma for providing me with accommodation and transportation during my internship at Authentec.

The PRIP lab is an excellent place to work in and it is one place where you can always find good company, no matter what time of the day it is. My interactions with members of this lab has certainly made me a better professional. Special thanks goes to Dr. Umut Uludag for helping me acclimatize to the lab during the initial months. I would also like to thank Dr. Xiaoguang Lu and Unsang Park for their assistance in the soft biometrics project, Yi Chen for sharing the joys and frustrations in research, Dr. Hong Chen for all his help during the NEC internship and Dr. Martin Law who was always ready to help in case of any technical problems. Finally, I would like to express my gratitude to other contemporary Prippies Dr. Anoop Namboodiri, Dr. Dirk Colbry, Yongfang Zhu, Meltem Demirkus, Steve Krawczyk, Jung-Eun Lee, Pavan Kumar Mallapragada, Abhishek Nagar, Leonardo Max Batista Claudino, and Miguel Figueroa-Villanue for creating and sustaining a lively and enjoyable research environment during my five years of stay in the windowless PRIP lab.

As a person who had never left the comforts of my home and family until I landed at Michigan State University, $2,000$ days of pleasant and unforgettable life in Michigan would not have been possible without the company of many great friends. First of all, I would like to sincerely thank my roommates Arun Prabakaran, Mahesh Arumugam, Srikanth Sridhar and Sriram Raghunath for their deep camaraderie, emo-

tional support and logistical help. In particular, the idea of doing a PhD with Dr. Jain would have never crossed by mind but for Mahesh, who had constantly extolled the virtues of this path and convinced me to follow it. I owe this thesis to him for all his guidance and help. I am also grateful to Sunil Unikkat, Shankarshna Madhavan, Narasimhan Swaminathan, Aravindhan Ravisekar, Prasanna Balasundaram, Loganathan Anjaneyulu, Raja Ganjikunta, Bruhadeswar Bezawada, Madhusudhan Srinivasan, Sudharson Sundararajan, Senthil Kumar Venkatesan and Amit Gore for all the fun-time we had together in Michigan. I am also very grateful to my friends Mahadevan Balakrishnan, Jayaram Venkatesan, Hariharan Rajasekaran, Balasubramanian Rathakrishnan and Balaji Arcot Srinivasan for their long conversations over phone that helped me feel at home.

Finally, I would like to thank my parents who have been the pillars of strength in all my endeavors. I am always deeply indebted to them for all that they have given me. I also thank all the other members of my family including my brother and two sisters for their love, affection and timely help.

# Table of Contents

# List of Tables

# List of Figures

Images in this dissertation are presented in color.

# Chapter 1

# Introduction

Personal identity refers to a set of attributes (e.g., name, social security number, etc.) that are associated with a person. Identity management is the process of creating, maintaining and destroying identities of individuals in a population. A reliable identity management system is urgently needed in order to combat the epidemic growth in identity theft and to meet the increased security requirements in a variety of applications ranging from international border crossing to accessing personal information. Establishing (determining or verifying) the identity of a person is called *person recognition* or *authentication* and it is a critical task in any identity management system. The three basic ways to establish the identity of a person are "something you know" (e.g., password, personal identification number), "something you carry" (e.g., physical key, ID card) and "something you are" (e.g., face, voice) [44].

Surrogate representations of identity such as passwords and ID cards can be easily misplaced, shared or stolen. Passwords can also be easily guessed using social engineering [136] and dictionary attacks [110]. Hence, the effective security provided

by passwords is significantly less than the expected security. Studies by the National Institute of Standards and Technology (NIST) [18] have estimated that on average, an 8-character ASCII (7 bits/character) password effectively provides only 18 bits of entropy, which is much less than the expected 56 bits of security. Moreover, passwords and ID cards cannot provide vital authentication functions like non-repudiation and detecting multiple enrollments. For example, users can easily deny using a service by claiming that their password has been stolen or guessed. Individuals can also conceal their true identity by presenting forged or duplicate identification documents. Therefore, it is becoming increasingly apparent that knowledge-based and token-based mechanisms alone are not sufficient for reliable identity determination and stronger authentication schemes based on "something you are", namely biometrics, are needed.

## 1.1 Biometric Systems

Biometric authentication, or simply biometrics, offers a natural and reliable solution to the problem of identity determination by establishing the identity of a person based on "who he is", rather than "what he knows" or "what he carries" [84]. Biometric systems automatically determine or verify a person's identity based on his anatomical and behavioral characteristics such as fingerprint, face, iris, voice and gait. Biometric traits constitute a strong and permanent "link" between a person and his identity and these traits cannot be easily lost or forgotten or shared or forged. Since biometric systems require the user to be present at the time of authentication, it can also deter users from making false repudiation claims. Moreover, only biometrics can provide

negative identification functionality where the goal is to establish whether a certain individual is indeed enrolled in the system although the individual might deny it. Due to these reasons, biometric systems are being increasingly adopted in a number of government and civilian applications either as a replacement for or to complement existing knowledge and token-based mechanisms. Some of the large scale biometric systems include the Integrated Automated Fingerprint Identification System (IAFIS) of the FBI [150], the US-VISIT IDENT program [149], the Schiphol Privium scheme at Amsterdam's Schiphol airport [176] and the finger scanning system at Disney World, Orlando [77].

A number of anatomical and behavioral body traits can be used for biometric recognition (see Figure 1.1). Examples of anatomical traits include face, fingerprint, iris, palmprint, hand geometry and ear shape. Gait, signature and keystroke dynamics are some of the behavioral characteristics that can be used for person authentication. Voice can be considered either as an anatomical or as a behavioral trait because certain characteristics of a person's voice such as pitch, bass/tenor and nasality are due to physical factors like vocal tract shape, and other characteristics such as word or phoneme pronunciation (e.g., dialect), use of characteristic words or phrases and conversational styles are mostly learned. Ancillary characteristics such as gender, ethnicity, age, eye color, skin color, scars and tatoos also provide some information about the identity of a person. However, since these ancillary attributes do not provide sufficient evidence to precisely determine the identity, they are usually referred to as soft biometric traits [89]. Each biometric trait has its advantages and limitations, and no single trait is expected to effectively meet all the requirements such as

accuracy, practicality and cost imposed by all applications [99]. Therefore, there is no universally best biometric trait and the choice of biometric depends on the nature and requirements of the application.

A typical biometric system consists of four main components, namely, sensor, feature extractor, matcher and decision modules. A sensor is used to acquire the biometric data from an individual. A quality estimation algorithm is sometimes used to ascertain whether the acquired biometric data is good enough to be processed by the subsequent components. When the data is not of sufficiently high quality, it is usually re-acquired from the user. The feature extractor gleans only the salient information from the acquired biometric sample to form a new representation of the biometric trait, called the *feature set*. Ideally, the feature set should be unique for each person (*extremely small inter-user similarity*) and also invariant with respect to changes in the different samples of the same biometric trait collected from the same person (*extremely small intra-user variability*). The feature set obtained during enrollment is stored in the system database as a *template*. During authentication, the feature set extracted from the biometric sample (known as *query* or *input* or *probe*) is compared to the template by the matcher, which determines the degree of similarity (dissimilarity) between the two feature sets. The decision module decides on the identity of the user based on the degree of similarity between the template and the query.

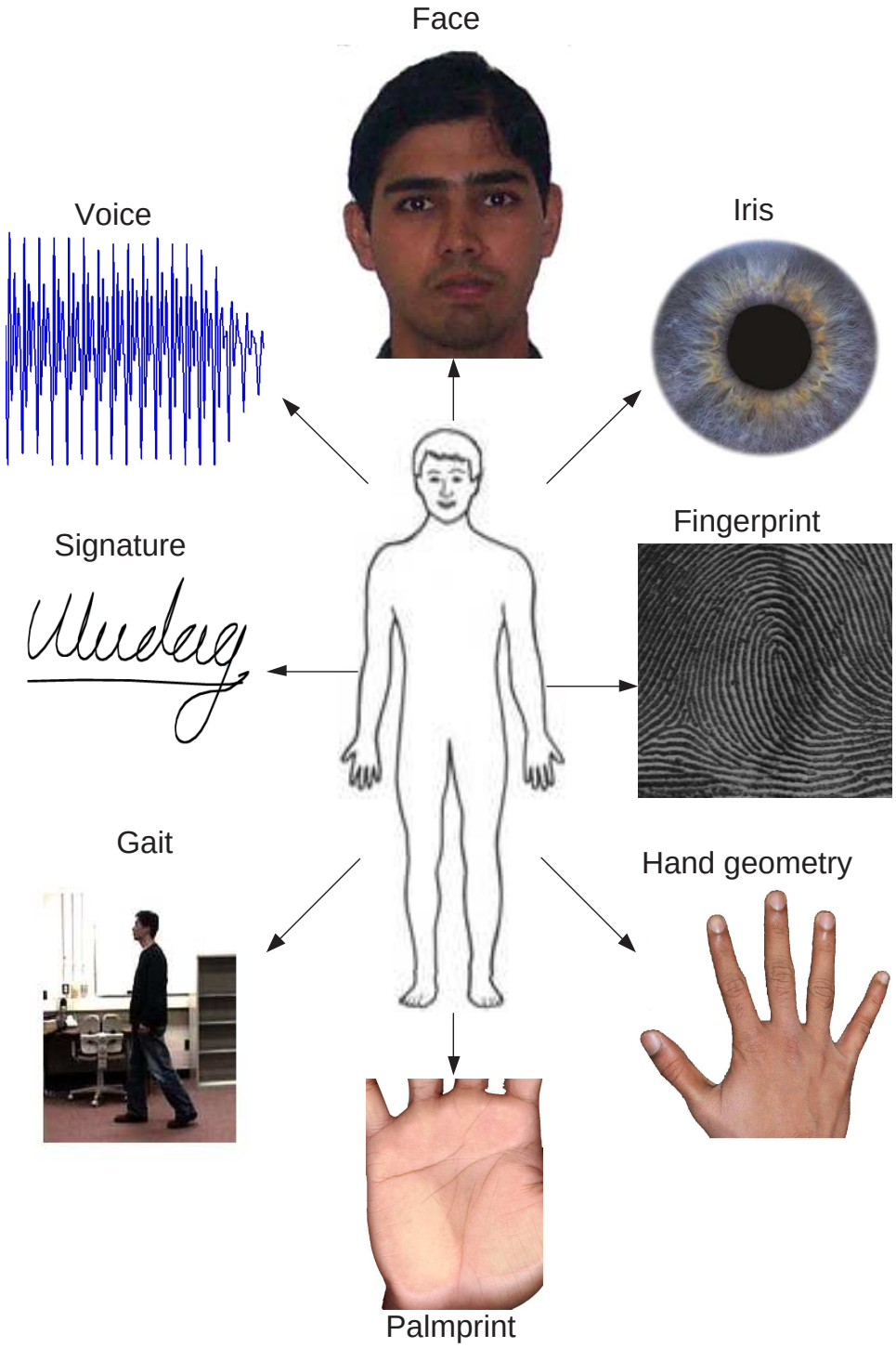



Figure 1.1: Examples of body traits that can be used for biometric recognition. Anatomical traits include face, fingerprint, iris, palmprint, hand geometry and ear shape, while gait, signature and keystroke dynamics are some of the behavioral characteristics. Voice can be considered either as an anatomical or as a behavioral characteristic.

## 1.2 Biometric Functionalities

The functionalities provided by a biometric system can be categorized[1] as *verification* and *identification*. Figure 1.2 shows the enrollment and authentication stages of a biometric system operating in the verification and identification modes. In verification, the user claims an identity and the system verifies whether the claim is genuine, i.e., the system answers the question "Are you who you say you are?". In this scenario, the query is compared only to the template corresponding to the claimed identity. If the user's input and the template of the claimed identity have a high degree of similarity, then the claim is accepted as "genuine". Otherwise, the claim is rejected and the user is considered an "impostor". Formally, verification can be posed as the following two-category classification problem: given a claimed identity $I$ and a query feature set $X_Q$, we need to decide if $(I, X_Q)$ belongs to "genuine" or "impostor" class. Let $X_I$ be the stored template corresponding to identity $I$. Typically, $X_Q$ is compared with $X_I$ and a *match score* $S$, which measures the similarity between $X_Q$ and $X_I$, is computed. The decision rule is given by

$$(I, X_Q) \in \begin{cases} \text{genuine}, & \text{if } S \geq \eta, \\ \text{impostor}, & \text{if } S < \eta, \end{cases} \tag{1.1}$$

where $\eta$ is a pre-defined threshold. In this formulation, the match score $S$ is assumed to measure the similarity between $X_Q$ and $X_I$, i.e., a large score indicates a good match. It is also possible for the match score to be a dissimilarity or distance measure

[1]Throughout this dissertation, the terms recognition or authentication will be used interchangeably when we do not wish to make a distinction between the verification and identification functionalities.

(i.e., a large score indicates a poor match) and in this case, the inequalities in the decision rule shown in equation (1.1) should be reversed.

Identification functionality can be classified into positive and negative identification. In positive identification, the user attempts to positively identify himself to the system without explicitly claiming an identity. A positive identification system answers the question "Are you someone who is known to the system?" by determining the identity of the user from a known set of identities. In contrast, the user in a negative identification application is considered to be concealing his true identity from the system. Negative identification is also known as screening and the objective of such systems is to find out "Are you who you say you are not?". Screening is often used at airports to verify whether a passenger's identity matches with any person on a "watch-list". Screening can also be used to prevent the issue of multiple credential records (e.g., driver's licence, passport) to the same person. Negative identification is also critical in applications such as welfare disbursement to prevent a person from claiming multiple benefits (i.e., double dipping) under different names. In both positive and negative identification, the user's biometric input is compared with the templates of all the persons enrolled in the database and the system outputs either the identity of the person whose template has the highest degree of similarity with the user's input or a decision indicating that the user presenting the input is not an enrolled user.

Formally, the problem of identification can be stated as follows: given a query feature set $X_Q$, we need to decide the identity $I$ of the user, where $I \in \{I_1, I_2, \cdots, I_N, I_{N+1}\}$. Here, $I_1, I_2, \cdots, I_N$ correspond to the identities of the $N$

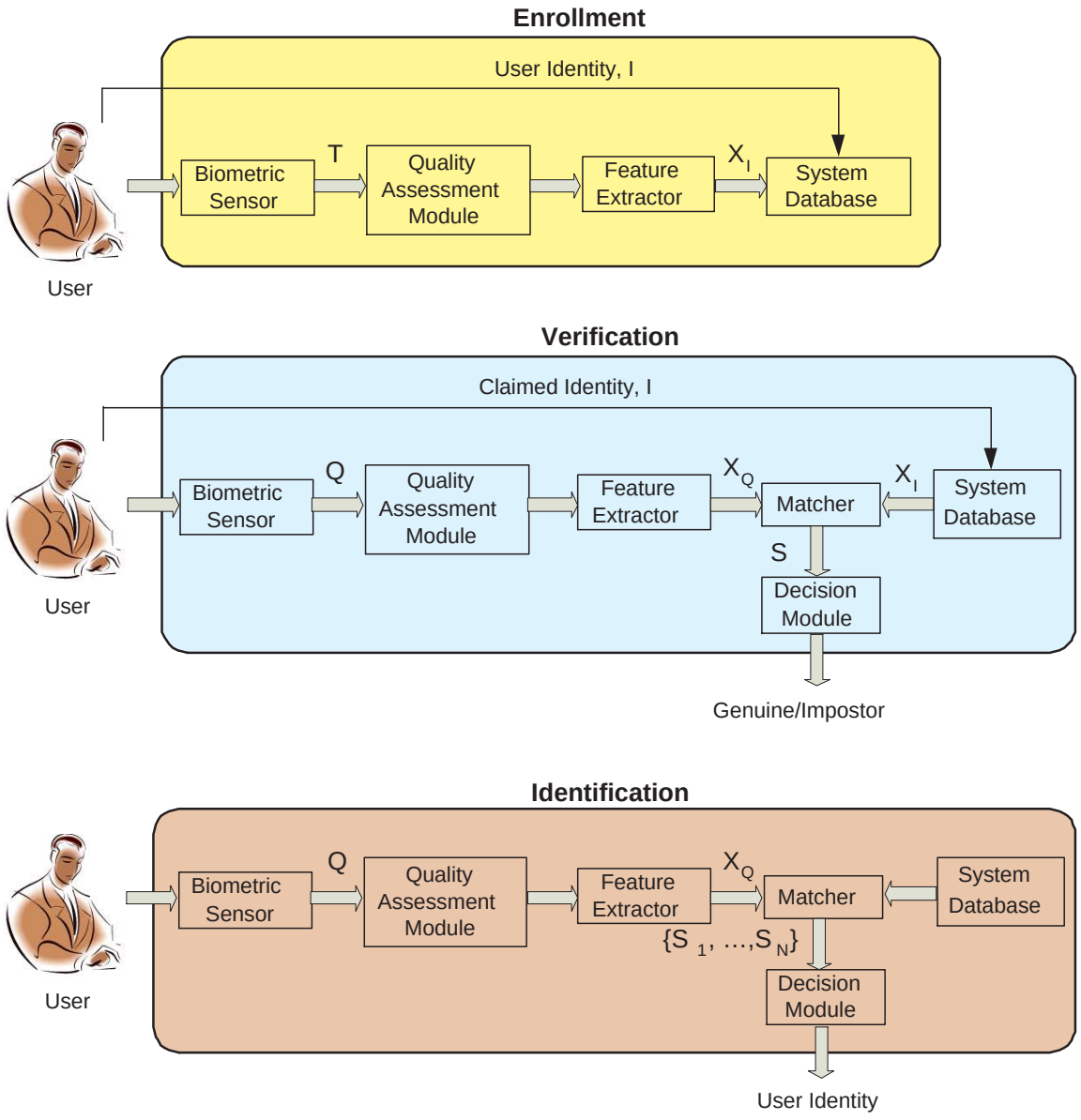



Figure 1.2: Enrollment and recognition stages in a biometric system. Here, $T$ represents the biometric sample obtained during enrollment, $Q$ is the query biometric sample obtained during recognition, $X_I$ and $X_Q$ are the template and query feature sets, respectively, $S$ represents the match score and $N$ is the number of users enrolled in the database.

users enrolled in the system and $I_{N+1}$ indicates the case where no suitable identity can be determined for the given query. If $X_{I_n}$ is the stored template corresponding to identity $I_n$ and $S_n$ is the match (similarity) score between $X_Q$ and $X_{I_n}$, for $n = 1, 2, \cdots, N$, the decision rule for identification is,

$$X_Q \in \begin{cases} I_{n_0}, & \text{if } n_0 = \arg\max_n S_n \text{ and } S_{n_0} \geq \eta, \\ I_{N+1}, & \text{otherwise,} \end{cases} \tag{1.2}$$

where $\eta$ is a pre-defined threshold. In some practical biometric identification systems such as FBI-IAFIS, identification is semi-automated, i.e., the biometric system outputs the identities of the top $m$ matches ($1 < m \ll N$) and a human expert manually determines the identity (among the $m$ selected identities) that best matches the given query. Note that the number of enrolled users in the database can be quite large. For example, there are more than 80 million subjects in the FBI-IAFIS system [150]. The presence of large number of identities in the database makes the identification task significantly more challenging than verification.

## 1.3 Performance of a Biometric System

Samples of the same biometric trait of a user obtained over a period of time can differ dramatically. The variability observed in the biometric feature set of an individual is known as *intra-user variations*. For example, in the case of fingerprints, factors such as placement of finger on the sensor, applied finger pressure, skin condition and feature extraction errors lead to large intra-user variations [129]. Figure 1.3 shows

two impressions of the same finger obtained on different days. Note how these impressions differ with respect to translation, rotation and non-linear distortion. On the other hand, features extracted from biometric traits of different individuals can be quite similar. For example, some pairs of individuals can have nearly identical facial appearance due to genetic factors (e.g., father and son, identical twins, etc.). Appearance-based facial features will exhibit a large similarity for these pairs of individuals and such a similarity is usually referred to as *inter-user similarity*.

A biometric system can make two types of errors, namely, false non-match and false match. When the intra-user variation is large, two samples of the same biometric trait of an individual (mate samples) may not be recognized as a match and this leads to a false non-match error. A false match occurs when two samples from different individuals (non-mate samples) are incorrectly recognized as a match due to large inter-user similarity. Therefore, the basic measures of the accuracy of a biometric system are *False Non-Match Rate* (FNMR) and *False Match Rate* (FMR). FNMR refers to the fraction of matches between two mate samples that are not recognized as a match and FMR is the proportion of matches between two non-mate samples that are incorrectly recognized as a match.

A False Non-Match Rate of 5% indicates that on average, 5 in 100 genuine attempts do not succeed. A majority of the false non-match errors are usually due to incorrect interaction of the user with the biometric sensor and can be easily rectified by allowing the user to present his/her biometric trait again. This is similar to the case where the user in a password-based authentication system makes a mistake while entering a password and is allowed to reenter the password.

Figure 1.3: Illustration of biometric intra-class variability. Two different impressions of the same finger obtained on different days are shown with minutia points marked on them. Due to differences in finger placement and distortion introduced by finger pressure variations, the number and location of minutiae in the two images are different (33 and 26 in the left and right images, respectively). The number of corresponding/matching minutiae in the two images is only 16 and some of these correspondences have been indicated in the figure.

A False Match Rate of 0.01% indicates that on average, 1 in 10,000 impostor attempts are likely to succeed. However, it must be emphasized that the security of a biometric system operating at 0.01% FMR is not equivalent to the security provided by a 4-digit PIN due to three reasons. Firstly, the adversary has to guess input values in the biometric feature space, which is requires significantly more effort and domain knowledge (e.g., knowledge about the features used in a particular biometric system, the statistical distribution of the features, the format of the stored templates, etc.) than what is required for guessing a PIN. Secondly, even if the adversary guesses the feature values, he must circumvent a physical component in the biometric system (sensor, feature extractor, or communication channels) in order to input the guessed features. This circumvention can be made very difficult by securing the physical infrastructure of the biometric system through appropriate techniques such as liveness detection, secure code execution and cryptographic protocols. Finally, it should be noted that the effective security provided by a 4-digit PIN is typically much less than 1 success in 10,000 impostor attempts, because most users tend to use numbers that are easy to remember (e.g., 1234, year of birth, etc.) and such PINs can be easily guessed by the adversary in a few attempts.

Apart from false non-match and false match, two other types of failures are also possible in a practical biometric system. If an individual cannot interact correctly with the biometric user interface or if the biometric samples of the individual are of very poor quality, the sensor or feature extractor may not be able to process these individuals. Hence, they cannot be enrolled in the biometric system and the proportion of individuals who cannot be enrolled is referred to as *Failure to Enroll Rate*

(FTER). In some cases, a particular sample provided by the user during authentication cannot be acquired or processed reliably. This error is called failure to capture and the fraction of authentication attempts in which the biometric sample cannot be captured is known as *Failure to Capture Rate* (FTCR).

In the context of biometric verification, FNMR and FMR are also known as False Reject Rate (FRR) and False Accept Rate (FAR), respectively. A match score is termed as *genuine* or *authentic* score if it indicates the similarity between two mate samples. An *impostor* score measures the similarity between two non-mate samples. As discussed in section 1.2, a verification system makes a decision by comparing the match score $S$ to a threshold $\eta$. Therefore, FRR can be defined as the proportion of genuine scores that are less than the threshold $\eta$ and FAR can be defined as the fraction of impostor scores that are greater than or equal to $\eta$. Let $f_{gen}(s) = p(S = s|genuine)$ and $f_{imp}(s) = p(S = s|impostor)$ be the probability density functions of the genuine and impostor scores, respectively. The FAR and FRR of the biometric system are given by

$$FAR(\eta) = p(S \geq \eta|impostor) = \int_{\eta}^{\infty} f_{imp}(s)ds, \tag{1.3}$$

$$FRR(\eta) = p(S < \eta|genuine) = \int_{-\infty}^{\eta} f_{gen}(s)ds. \tag{1.4}$$

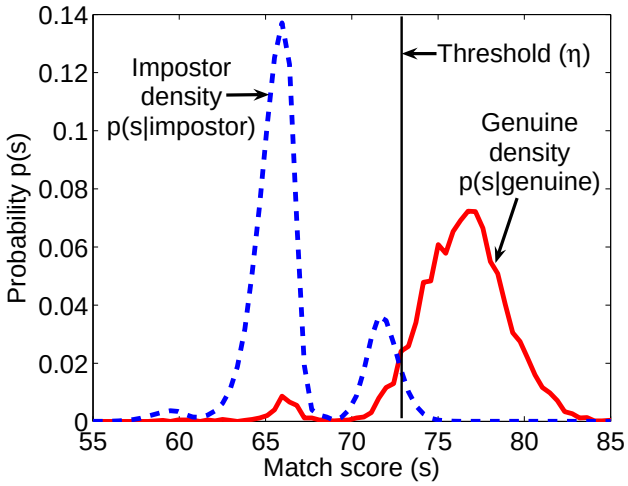
Both FRR and FAR are functions of the system threshold $\eta$. If the threshold is increased, FAR will decrease but the FRR will increase and vice versa. Hence, for a given biometric system, it is not possible to decrease both these errors simultane-

ously by varying the threshold. The Genuine Accept Rate (GAR) can be used as an alternative to FRR while reporting the performance of a biometric verification system. GAR is defined as the fraction of genuine scores that exceed the threshold $\eta$. Therefore,

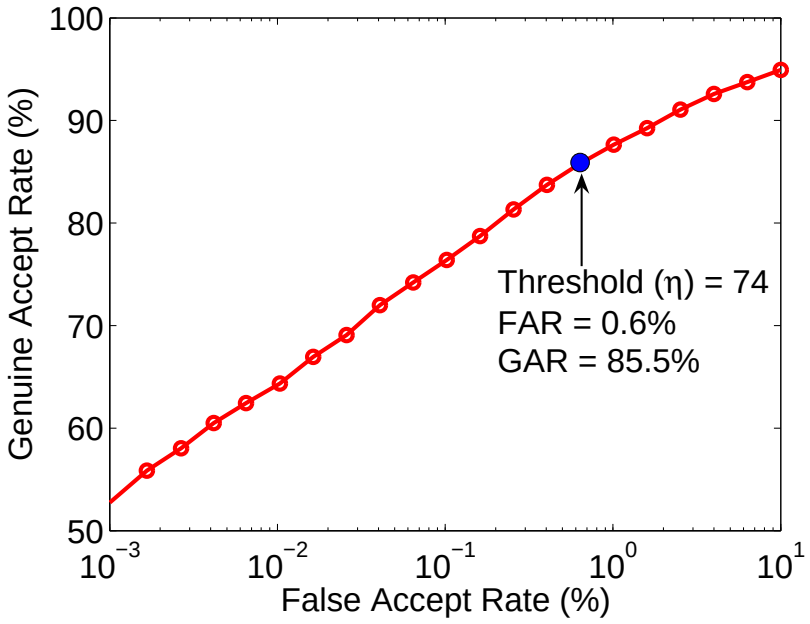
$$GAR(\eta) = p(S \geq \eta | genuine) = 1 - FRR(\eta). \tag{1.5}$$

The FAR and FRR of a biometric verification system at different values of threshold $\eta$ can be summarized in the form of a Detection Error Tradeoff (DET) or Receiver Operating Characteristic (ROC) curve. While the DET plot uses the normal deviate scale, ROC curves are plotted in a linear, semi-logarithmic or logarithmic scale. Equal Error Rate (EER) is the point in a DET or ROC curve where the FAR equals the FRR. A lower EER value indicates better performance. In this dissertation, we plot the ROC curve (GAR against FAR) on the semi-logarithmic scale to summarize the verification performance. Figure 1.4(a) shows the genuine and impostor score densities of the Face-G matcher in the NIST-BSSR1 database [151] and figure 1.4(b) shows the corresponding ROC curve.

The performance of a biometric identification system is measured in terms of the *identification rate*. Identification rate is the proportion of times the identity determined by the system is the true identity of the user providing the query biometric sample. If the biometric system outputs the identities of the top $m$ matches, the rank-$m$ identification rate, $R_m$, is defined as the proportion of times the true identity of the user is contained in the top $m$ matching identities. The identification rate at




(a)




(b)

Figure 1.4: Performance of a biometric system operating in the verification mode. (a) The genuine and impostor match score densities corresponding to the Face-G matcher in the NIST BSSR1 database. The threshold, $\eta$, determines the FAR and GAR of the system. (b) Receiver operating characteristic (ROC) curve for the Face-G matcher which plots the GAR against FAR on a semi-logarithmic scale.

different ranks can be summarized using the Cumulative Match Characteristic (CMC) curve [139] (see Figure 1.5), which plots $R_m$ against $m$ for $m = 1, 2, \cdots, N$, where $N$ is the number of enrolled users. When the same matcher is used for both verification and identification, then the corresponding ROC and CMC curves are related and the CMC curve can be estimated from the genuine and impostor score densities $f_{gen}(s)$ and $f_{imp}(s)$ [12, 75].

## 1.4 Challenges in Biometrics

Though biometric systems have been successfully deployed in a number of real-world applications, biometrics is not yet a fully solved problem. The three main factors that contribute to the complexity of biometric system design are accuracy (FAR, GAR and rank-1 identification rate), scalability (size of the database) and usability (ease of use, security and privacy). Jain et al. [92] state that the grand challenge in biometrics is to design a system that operates in the extremes of all these three factors. In other words, the challenge is to develop a biometric system that is highly accurate and secure, convenient to use and easily scalable to a large population. We now discuss the major obstacles that hinder the design of such an "ideal" biometric system.

### 1.4.1 Accuracy

An ideal biometric system should always provide the correct identity decision when a biometric sample is presented. However, a biometric system seldom encounters a

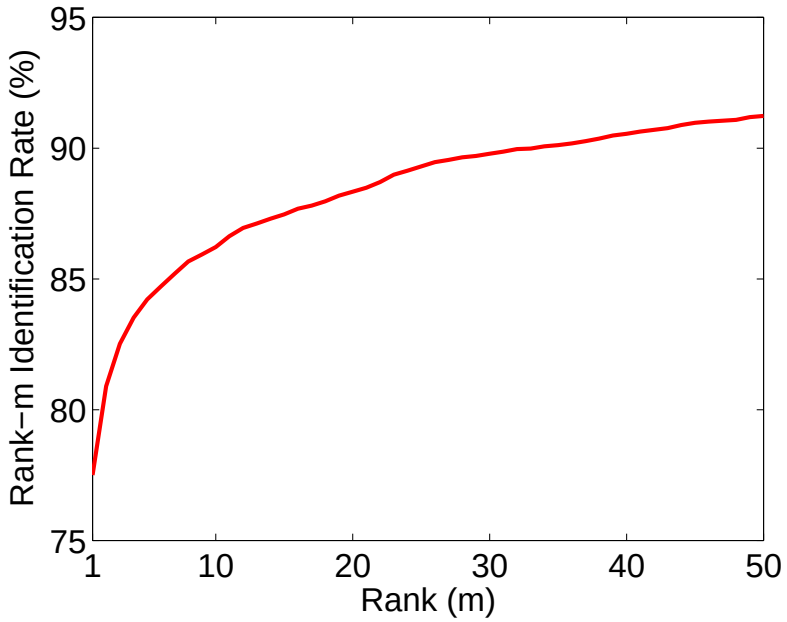
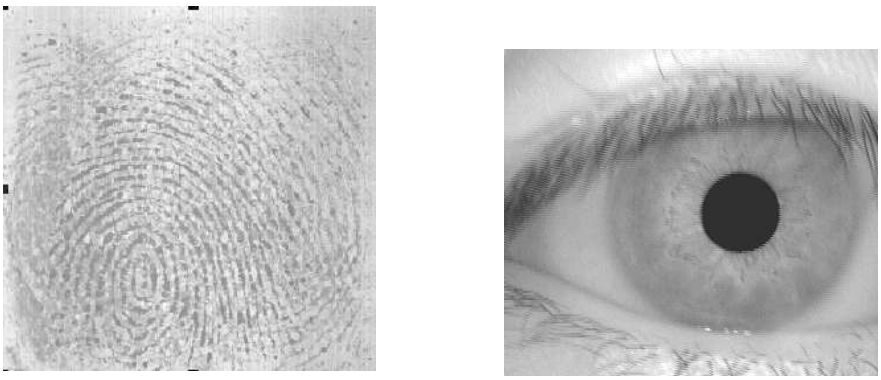



Figure 1.5: Cumulative match characteristic (CMC) curve for the Face-G matcher in the NIST BSSR1 database which plots the rank-$m$ identification rate for various values of $m$. In this example, the rank-1 identification rate is $\approx 78\%$ which means that for $\approx 78\%$ of the queries, the true identity of the query user is selected as the best matching identity.

sample of a user's biometric trait that is exactly the same as the template. This results in a number of errors as discussed in section 1.3 and thereby limits the system accuracy. The main factors affecting the accuracy of a biometric system [97] are:



(a) (b)

Figure 1.6: Examples of noisy biometric data; (a) A noisy fingerprint image due to smearing, residual deposits, etc.; (b) A blurred iris image due to loss of focus.

(a) (b) (c)

Figure 1.7: Non-universality of a biometric trait. This figure shows three impressions of a user's finger in which the ridge details are worn-out.

- Noisy sensor data: Noise can be present in the acquired biometric data mainly due to defective or improperly maintained sensors. For example, accumulation

of dirt or the residual remains on a fingerprint sensor can result in a noisy fingerprint image as shown in Figure 1.6(a). Failure to focus the camera appropriately can lead to blurring in face and iris images (see Figure 1.6(b)). The recognition accuracy of a biometric system is highly sensitive to the quality of the biometric input and noisy data can result in a significant reduction in the GAR of a biometric system [72, 204].

- Non-universality: If every individual in the target population is able to present the biometric trait for recognition, then the trait is said to be universal. Universality is one of the basic requirements for a biometric identifier. However, not all biometric traits are truly universal. The National Institute of Standards and Technology (NIST) has reported that it is not possible to obtain a good quality fingerprint from approximately two percent of the population (people with hand-related disabilities, manual workers with many cuts and bruises on their fingertips, and people with very oily or dry fingers) [189] (see Figure 1.7). Hence, such people cannot be enrolled in a fingerprint verification system. Similarly, persons having long eye-lashes and those suffering from eye abnormalities or diseases like glaucoma, cataract, aniridia, and nystagmus cannot provide good quality iris images for automatic recognition [147]. Non-universality leads to high FTER and FTCR in a biometric system.

- Inter-user similarity: Inter-user similarity refers to the overlap of the biometric samples from two different individuals in the feature space. The lack of uniqueness in the biometric feature set restricts the discriminative ability of the

biometric system and leads to an increase in the FMR. In the case of a biometric identification system, the inherent information constraint in the feature set results in an upper bound on the number of unique individuals that can be accommodated.

- Lack of invariant representation: Biometric samples of an individual usually exhibit large intra-user variations (see Figure 1.3). The variations may be due to improper interaction of the user with the sensor (e.g., changes due to rotation, translation and applied pressure when the user places his finger on a fingerprint sensor, changes in pose and expression when the user stands in front of a camera, etc.), use of different sensors during enrollment and verification, changes in the ambient environmental conditions (e.g., illumination changes in a face recognition system) and inherent changes in the biometric trait (e.g., appearance of wrinkles due to aging or presence of facial hair in face images, presence of scars in a fingerprint, etc.). Ideally, the features extracted from the biometric data must be relatively invariant to these changes. However, in most practical biometric systems the features are not invariant and therefore complex matching algorithms are required to take these variations into account. Large intra-user variations usually decrease the GAR of a biometric system.

Due to the above factors, the error rates associated with biometric systems are higher than what is required in many applications. Table 1.1 summarizes the error rates of fingerprint, face, iris and voice biometric systems obtained through various technology evaluation tests. Although the error rates presented in Table 1.1 are

dependent on a number of test conditions such as the sensor used, the acquisition protocol, the number and demographic profile of the subjects involved and the time lapse between successive biometric acquisitions, they provide a good estimate of the accuracy of state-of-the-art unibiometric systems because these results are obtained by independent third-party testing of competing algorithms on common databases. The results of these evaluations clearly indicate that biometric systems have non-zero error rates and there is scope for improving the accuracy of biometric systems.

Table 1.1: False reject and false accept rates associated with state-of-the-art fingerprint, face, voice and iris verification systems. Note that the accuracy estimate of a biometric system depends on a number of test conditions.

| **Biometric Trait** | **Test** | **Test Conditions** | **False Reject Rate** | **False Accept Rate** |
|---|---|---|---|---|
| **Fingerprint** | FVC 2006 [148] | Heterogeneous population including manual workers and elderly people | 2.2% | 2.2% |
| | FpVTE 2003 [204] | U.S. government operational data | 0.1% | 1% |
| **Face** | FRVT 2006 [153] | Controlled illumination, high resolution | 0.8-1.6% | 0.1% |
| **Voice** | NIST 2004 [156] | Text independent, multi-lingual | 5-10% | 2-5% |
| **Iris** | ICE 2006 [153] | Controlled illumination, broad quality range | 1.1-1.4% | 0.1% |

### 1.4.2 Scalability

In the case of a biometric verification system, the size of the database (number of enrolled users in the system) is not an issue because each authentication attempt

basically involves matching the query with a single template. In the case of large scale identification systems where $N$ identities are enrolled in the system, sequentially comparing the query with all the $N$ templates is not an effective solution due to two reasons. Firstly, the throughput[2] of the system would be greatly reduced if the value of $N$ is quite large. For example, if the size of the database is 1 million and if each match requires an average of 100 microseconds, then the throughput of the system will be less than 1 per minute. Furthermore, the large number of identities also affects the false match rate of the system adversely. Hence, there is a need for efficiently scaling the system. This is usually achieved by a process known as filtering or indexing where the database is pruned based on extrinsic (e.g., gender, ethnicity, age, etc.) or intrinsic (e.g., fingerprint pattern class) factors and the search is restricted to a smaller fraction of the database that is likely to contain the true identity of the user. There are very few published studies on efficiently indexing biometric databases [9, 52, 73] and this is still an active area of research in the biometrics community.

### 1.4.3 Security and Privacy

Although it is difficult to steal someone's biometric traits, it is still possible for an impostor to circumvent a biometric system in a number of ways [160]. For example, it is possible to construct fake or spoof fingers using lifted fingerprint impressions (e.g., from the sensor surface) and utilize them to circumvent a fingerprint recognition system [133, 134]. Behavioral traits like signature [78] and voice [58] are more

[2]Throughput of a biometric system is defined as the number of queries (authentication attempts) that can be processed per unit time.

susceptible to such attacks than anatomical traits.

The most straightforward way to secure a biometric system is to put all the system modules and the interfaces between them on a smart card (or more generally a secure processor). In such systems, known as match-on-card or system-on-card technology, sensor, feature extractor, matcher and template reside on the card [91]. The advantage of this technology is that the user's biometric data never leaves the card which is in the user's possession. However, system-on-card solutions are not appropriate for most large-scale verification applications because they are still expensive and users must carry the card with them at all times. Moreover, system-on-card solutions cannot be used in identification applications.

One of the critical issues in biometric systems is protecting the template of a user which is typically stored in a database or a smart card. Stolen biometric templates can be used to compromise the security of the system in the following two ways. (i) The stolen template can be replayed to the matcher to gain unauthorized access, and (ii) a physical spoof can be created from the template (see [2,21,171]) to gain unauthorized access to the system (as well as other systems which use the same biometric trait). Note that an adversary can covertly acquire the biometric information of a genuine user (e.g., lift the fingerprint from a surface touched by the user). Hence, spoof attacks are possible even when the adversary does not have access to the biometric template. However, the adversary needs to be in the physical proximity of the person he is attempting to impersonate in order to covertly acquire his biometric trait. On the other hand, even a remote adversary can create a physical spoof if he gets access to the biometric template information.

Unlike passwords, when biometric templates are compromised, it is not possible for a legitimate user to revoke his biometric identifiers and switch to another set of uncompromised identifiers. Due to this irrevocable nature of biometric data, an attack against the stored templates constitutes a major security and privacy threat in a biometric system.

Since a biometric trait is a permanent link between a person and his identity, it can be easily prone to abuse in such a way that a person's right to privacy and anonymity is compromised. A common type of abuse of biometric identifiers is *function creep* [84] where the acquired biometric identifiers are later used for purposes other than the intended purpose. For example, Disney World in Orlando collects fingerprints from park visitors in order to prevent customers from sharing the tickets with others [77]. However, it is possible that the same fingerprints may be used later for searching against a criminal fingerprint database or cross-link it to a person's health records. Hence, strategies to prevent function creep and to ensure an individual's privacy are urgently needed.

## 1.5 Summary

Biometric recognition is the process of establishing the identity of a person based on his anatomical or behavioral characteristics. Since biometric traits provide irrefutable evidence linking a person to his identity, biometric authentication is a natural and reliable solution to the problem of establishing the identity of an individual in any identity management system. While biometric systems offer a number of functional-

ities such as verification, positive identification and screening, these systems are not perfect. Due to factors like intra-user variations and inter-user similarity, the error rates associated with biometric systems is non-zero. Besides the accuracy, the high failure rates (FTER and FTCR), scalability and various vulnerabilities also limit the deployment of biometric systems in many applications. While rapid progress has been made in the development and deployment of biometric systems in the past few decades, a number of core research issues in biometrics have not yet been fully addressed.

Solutions to advance the state of the art in biometrics include the design of new sensors that can acquire the biometric traits of an individual in a more reliable, convenient and secure manner, the development of invariant representation schemes and robust and efficient matching algorithms, combining evidence from multiple biometric sources to compensate for the limitations of the individual sources and the development of techniques for liveness detection, template security and privacy enhancement of biometric systems. In this thesis, we focus on biometric systems that integrate cues obtained from multiple biometric sources and these systems are commonly referred to as multibiometric systems. Multibiometric systems offer a number of advantages that can alleviate the problems associated with traditional (uni)biometric systems. This thesis addresses two critical issues in the design of a multibiometric system, namely, fusion methodology and template security.

## 1.6 Thesis contributions

The first part of this dissertation addresses the problem of fusion in a multibiometric system and the second part deals with the problem of multibiometric template security. The major contributions of this dissertation are as follows.

- We propose a principled approach based on the likelihood ratio test for fusion of match scores from multiple biometric matchers in the verification scenario. The proposed fusion framework is based on the Neyman-Pearson theorem, which guarantees that at any specified FAR, the likelihood ratio test maximizes the GAR, provided the genuine and impostor match score densities are known. We use a semi-parametric density estimation approach, namely, finite Gaussian mixture models (GMM) to estimate the joint densities of match scores. We demonstrate that fusion based on these density estimates achieves consistently high performance on different multibiometric databases involving face, fingerprint, iris, and speech modalities. We also extend the likelihood ratio based fusion scheme to incorporate the quality of the biometric samples and define new quality metrics known as pairwise quality indices for fingerprint and iris images. We also propose a technique based on decision trees to design cascade multibiometric systems within the likelihood ratio framework.

- We investigate rank and score level fusion schemes in a multibiometric identification system and show that the genuine and impostor likelihood ratios used in the verification scenario can also be applied in the case of identification if we assume that the match scores of the individual users are independent and

identically distributed.

- We propose a feature level fusion scheme for securing multibiometric templates using the fuzzy vault framework. The proposed framework can handle multiple samples (e.g., two impressions from the same finger), multiple instances (e.g., impressions from left and right index fingers of a person) and multiple biometric traits (e.g., fingerprint and iris). Towards this end, we have developed a fully automatic implementation of a fingerprint-based fuzzy vault where helper data derived from the fingerprint orientation field is used to align the template and query minutiae. We have also developed an iris-based fuzzy vault for securing iriscode templates. Finally, we show that a multibiometric vault that utilizes multiple fingerprint impressions or multiple fingers or fingerprint and iris achieves better accuracy and security compared to a unibiometric vault.

# Chapter 2

# Multibiometric Systems

Systems that consolidate evidence from multiple sources of biometric information in order to reliably determine the identity of an individual are known as multibiometric systems [169]. Multibiometric systems can alleviate many of the limitations of unibiometric systems because the different biometric sources usually compensate for the inherent limitations of the other sources [81]. Multibiometric systems offer the following advantages over unibiometric systems.

1. Combining the evidence obtained from different sources using an effective fusion scheme can significantly improve the overall accuracy of the biometric system. The presence of multiple sources also effectively increases the dimensionality of the feature space and reduces the overlap between the feature spaces of different individuals.

2. Multibiometric systems can address the non-universality problem and reduce the FTER and FTCR. For example, if a person cannot be enrolled in a finger-

print system due to worn-out ridge details, he can still be identified using other biometric traits like face or iris.

3. Multibiometric systems can also provide a certain degree of flexibility in user authentication. Suppose a user enrolls into the system using several different traits. Later, at the time of authentication, only a subset of these traits may be acquired based on the nature of the application under consideration and the convenience of the user. For example, consider a banking application where the user enrolls into the system using face, voice and fingerprint. During authentication, the user can select which trait to present depending on his convenience. While the user can choose face or voice modality when he is attempting to access the application from his mobile phone equipped with a digital camera (see Figure 2.1), he can choose the fingerprint modality when accessing the same application from a public ATM or a network computer.

4. The availability of multiple sources of information considerably reduces the effect of noisy data. If the biometric sample obtained from one of the sources is not of sufficient quality during a particular acquisition, the samples from other sources may still provide sufficient discriminatory information to enable reliable decision-making.

5. Multibiometric systems can provide the capability to search a large database in a computationally efficient manner. This can be achieved by first using a relatively simple but less accurate modality to prune the database before using the more complex and accurate modality on the remaining data to perform

the final identification task. This will improve the throughput of a biometric identification system.

6. Multibiometric systems are more resistant to spoof attacks because it is difficult to simultaneously spoof multiple biometric sources. Further, a multibiometric system can easily incorporate a challenge-response mechanism during biometric acquisition by acquiring a subset of the traits in some random order (e.g, left index finger followed by face and then right index finger). Such a mechanism will ensure that the system is interacting with a live user. Further, it is also possible to improve the template security by combining the feature sets from different biometric sources using an appropriate fusion scheme.

Multibiometric systems also have a few disadvantages when compared to unibiometric systems. They are more expensive and require more resources for computation and storage than unibiometric systems. Multibiometric systems generally require additional time for user enrollment, causing some inconvenience to the user. Finally, the accuracy of a multibiometric system can actually be lower than that of the unibiometric system if an appropriate technique is not followed for combining the evidence provided by the different sources. Still, multibiometric systems offer features that are attractive and as a result, such systems are being increasingly deployed in security-critical applications (e.g., FBI-IAFIS [150], US-VISIT IDENT program [149], etc.).


Figure 2.1: A hypothetical mobile banking application where the user has the flexibility to choose all or a subset of available biometric traits (e.g., face, voice and fingerprint) for authentication depending on his convenience. Research is under way to perform iris recognition based on images captured using the camera on the mobile phone [100].

## 2.1 Design Issues in Multibiometrics

The design of a multibiometric system is dependent on the requirements of the application. The major issues that need to be considered in the design of a multibiometric system are described below.

1. Sources of biometric information include multiple sensors, multiple representations and matching algorithms, multiple samples of the same biometric trait, multiple instances of a biometric trait and multiple biometric traits. For a given application, the system designer needs to decide which of these sources should be used in designing the multibiometric system.

2. The sequence in which the multiple sources of information are acquired and processed could be serial (cascade or sequential), parallel or hierarchical (tree-like). Depending on the application scenario, an appropriate acquisition and processing architecture must be selected.

3. The process of integrating evidence provided by different biometric sources is known as biometric fusion. Four types of information can be obtained from the biometric sources, namely, raw biometric samples, feature sets, match scores and decision labels. Depending on the type of information that is fused, the fusion scheme can be classified as sensor level, feature level, score level and decision level fusion. The choice of the fusion level is the most important design issue in a multibiometric system and it has a substantial impact on the performance of the system.

4. Given the type of information to be fused, a number of techniques are available for fusion of information provided by the multiple sources. Many of these fusion schemes may be admissible in an application and the challenge is to find the optimal one.

It must be mentioned that a majority of the design decisions are based on a cost-benefit analysis. Typically, there is a tradeoff between the additional cost and the improvement in performance of a multibiometric system. The cost could be a function of the number of sensors deployed, the time required for acquisition and processing (throughput), performance gain (reduction in FAR/FRR), storage and computational requirements and perceived (in)convenience to the user.

## 2.2 Sources of Multiple Evidence

Sources of information in a multibiometric system (see Figure 2.2) may include (i) *multiple sensors* to capture the same biometric trait (e.g., face captured using optical and range sensors), (ii) *multiple representations* or *multiple algorithms* for the same biometric trait (e.g., texture and minutiae-based fingerprint matchers), (iii) *multiple instances* of the same biometric trait (e.g., left and right iris), (iv) *multiple samples* of the same biometric trait (e.g., two impressions of a person's right index finger), and (v) *multiple biometric traits* (e.g., face and iris).

In the first four scenarios, multiple sources of information are derived from the same biometric trait. In the fifth scenario, information is derived from different biometric traits and these systems are known as *multimodal* biometric systems. In fact,

biometric fusion can also be carried out on any arbitrary combination of the above five sources and such systems can be referred to as *hybrid* multibiometric systems [26]. An example of a hybrid multibiometric system is the system proposed by Brunelli et al. [15] where the results of two speaker recognition algorithms are combined with three face recognition algorithms at the match score and rank levels using a HyperBF network. Hence, this system is multi-algorithmic as well as multimodal in its design.

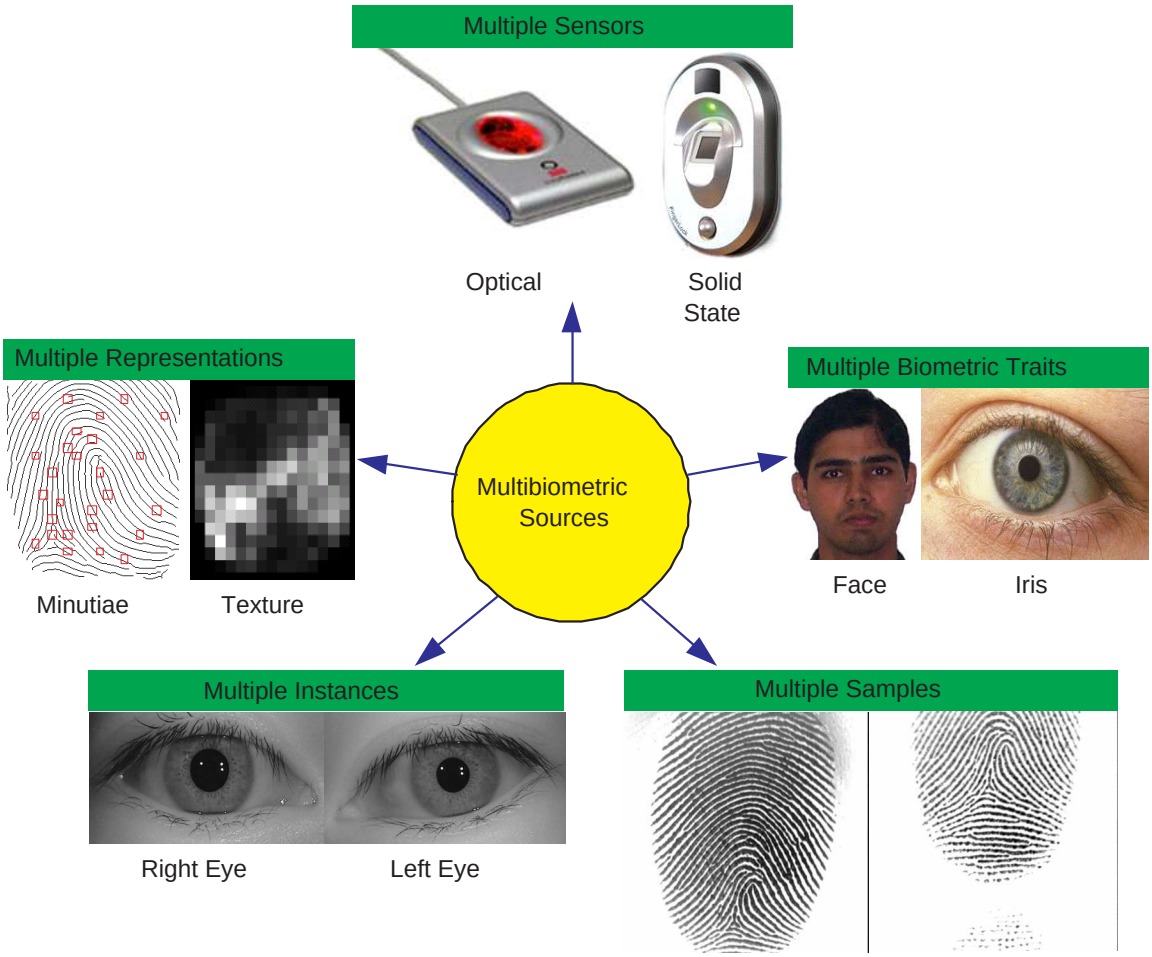


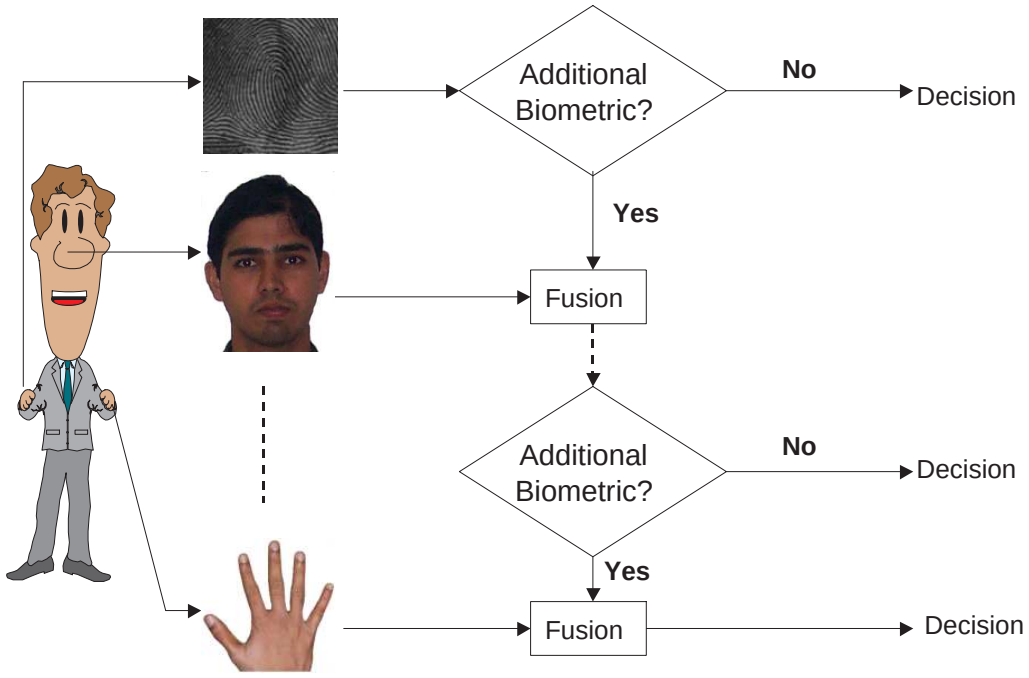

Figure 2.2: Various sources of information that can be fused in a multibiometric system. In four of the five scenarios (multiple sensors, representations, instances and samples), multiple sources of information are derived from the same biometric trait. In the fifth scenario, information is derived from different biometric traits and such systems are known as multimodal biometric systems.

## 2.3 Acquisition and Processing Sequence

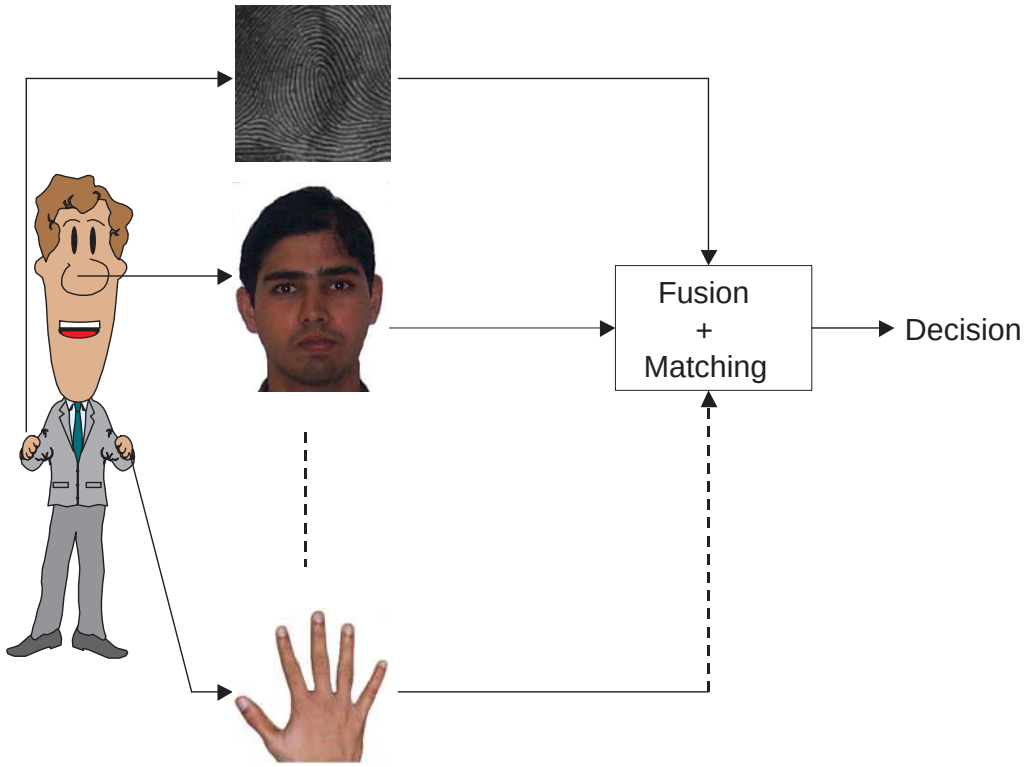
The order or sequence in which biometric samples are acquired and processed can have a significant impact on the time required for enrollment and authentication, failure to enroll rate (FTER) and user convenience. Typically, the acquisition and processing architecture of a multibiometric system is either serial or parallel (see Figure 2.3). In the serial or cascade or sequential architecture, the acquisition and processing of the different sources take place sequentially and the outcome of one matcher may affect the processing of the subsequent sources. In the parallel design, different sources are processed independently and their results are combined using an appropriate fusion scheme. Both these architectures have their own advantages and limitations.

In the case of biometric acquisition, both serial and parallel architectures are quite common. It is usually convenient and cost-effective to acquire physically related biometric traits simultaneously. For example, face, voice and lip movement can be simultaneously acquired using a video camera [69]. Similarly, palmprint and hand-geometry information can be acquired in parallel using a single camera [112]. On the other hand, when multiple instances of the same trait (e.g., iris images from both the eyes) or physically unrelated biometric traits (e.g., fingerprint and face) need to be acquired, the acquisition is usually done sequentially.

Most of the multibiometric systems proposed in the literature follow a parallel architecture for processing the biometric information. This is because the primary goal of system designers has been a reduction in the error rate of biometric systems and the parallel mode of processing generally has a higher accuracy because it utilizes




(a)




(b)

Figure 2.3: Acquisition and processing architecture of a multibiometric system; (a) Serial (Cascade or Sequential) and (b) Parallel.

more evidence about the user for recognition [167, 180]. However, a cascading architecture may have other advantages such as increased user convenience and higher throughput, which may be useful in large scale identification tasks. For example, when a cascaded multibiometric system has sufficient confidence on the identity of the user after processing the first biometric source, the user may not be required to provide the other sources of information. The system can also allow the user to decide which information source he/she would present first. Finally, if the system is faced with the task of identifying the user from a large database, it can utilize the outcome of each matcher to successively prune the database, thereby making the search faster and more efficient. Thus, a cascaded system can be more convenient to the user and generally requires less recognition time when compared to its parallel counterpart. An example of a cascaded multibiometric system is the one proposed by Hong and Jain [80]. In this system, face recognition is used to retrieve the top $m$ matching identities and fingerprint recognition is used to verify these identities and make a final identification decision.

The choice of the system architecture depends on the application requirements. User-friendly and low security applications like bank ATMs can use a cascaded multibiometric system. On the other hand, parallel multibiometric systems are more suited for applications where security is of paramount importance (e.g., access to military installations). It is also possible to design a hierarchical (tree-like) architecture to combine the advantages of both cascade and parallel architectures. This hierarchical architecture can be made dynamic so that it is robust and can handle problems like missing and noisy biometric samples that often arise in biometric systems [129]. How-

ever, the design of a hierarchical multibiometric system has not yet received adequate attention from researchers.

## 2.4 Levels of Fusion

One of the fundamental issues in the design of a multibiometric system is to determine the type of information that should be fused. Depending on the type of information that is fused, the fusion scheme can be classified as sensor level, feature level, score level and decision level fusion. Typically, the amount of information available to the system decreases as one proceeds from the sensor module to the decision module (see Figure 2.4). The raw biometric data (e.g., face image in the case of face biometric) has the highest information content, which gets reduced by subsequent processing (e.g., after extraction of PCA features). In the verification mode, the final decision label contains only a single bit of information (match or non-match). However, the different stages of biometric data processing are expected to decrease the intra-user variability and the amount of noise that is contained in the available information. Further, in many practical multibiometric systems, higher levels of information such as the raw images or feature sets are either not available (e.g., proprietary feature sets used in commercial-off-the-shelf systems) or the information available from different sources is not compatible (e.g., fingerprint minutiae and eigenface coefficients). On the other hand, in most of the multibiometric systems, it is relatively easy to access and combine the match scores generated by different biometric matchers. Therefore, information fusion at the match score level offers the best tradeoff in terms of information content

and ease in fusion. Consequently, score level fusion is the most commonly used approach in multibiometric systems.

Figure 2.5 shows examples of fusion at the various levels in a multibiometric system. The four levels of fusion can be broadly categorized as (i) fusion prior to matching and (ii) fusion after matching [173]. This distinction is made because once the biometric matcher is applied, the amount of information available to the system drastically decreases.

### 2.4.1 Fusion Prior to Matching

Prior to matching, integration of information from multiple biometric sources can take place either at the sensor level or at the feature level.

#### Sensor Level Fusion

The raw data from the sensor(s) are combined in sensor level fusion [83]. Sensor level fusion can be performed only if the sources are either samples of the same biometric trait obtained from multiple compatible sensors or multiple instances of the same biometric trait obtained using a single sensor. For example, multiple 2D face images obtained from different viewpoints can be stitched together to form a 3D model of the face [123] or a panaromic face mosaic [207]. Another example of sensor level fusion is the mosaicing of multiple fingerprint impressions to form a more complete fingerprint image [40, 95, 140, 159, 212]. In sensor level fusion, the multiple cues must be compatible and the correspondences between points in the raw data must be either known in advance (e.g., calibrated camera systems) or reliably estimated.

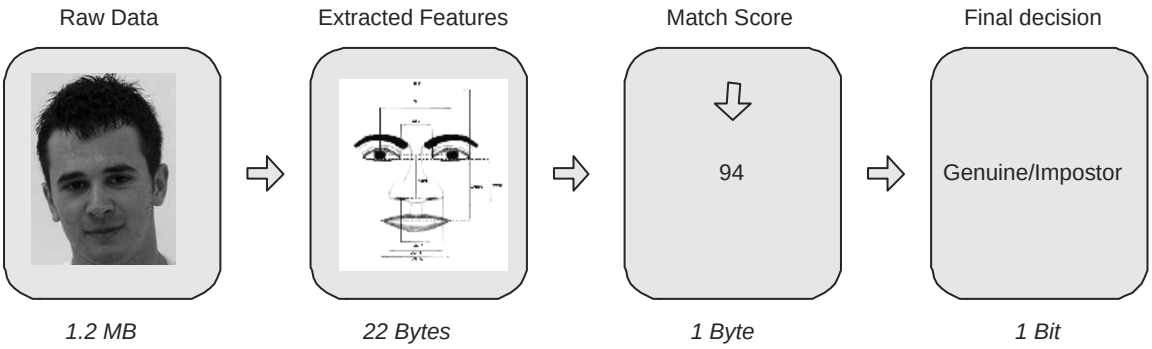



Figure 2.4: The amount of information available for fusion decreases progressively after each layer of processing in a biometric system. The raw data represents the richest source of information, while the final decision (in a verification scenario) contains just a single bit of information. However, the raw data is corrupted by noise and may have large intra-class variability, which is expected to be reduced in the subsequent modules of the system. (Reproduced from [169])

### Feature Level Fusion

Feature level fusion refers to combining different feature sets that are extracted from multiple biometric sources. When the feature sets are homogeneous (e.g., multiple fingerprint impressions of a user's finger), a single resultant feature set can be calculated as a weighted average of the individual feature sets (e.g., mosaicing of fingerprint minutiae [170]). When the feature sets are non-homogeneous (e.g., feature sets of different biometric modalities like face and hand geometry), we can concatenate them to form a single feature set. Feature selection schemes can then be applied to reduce the dimensionality of the resultant feature set [166]. Concatenation is not possible when the feature sets are incompatible (e.g., fingerprint minutiae and eigenface coefficients). When the multiple feature sets correspond to different samples of the same biometric trait that are processed using the same feature extraction algorithm, then feature level fusion can be considered as template update or template improvement [101].

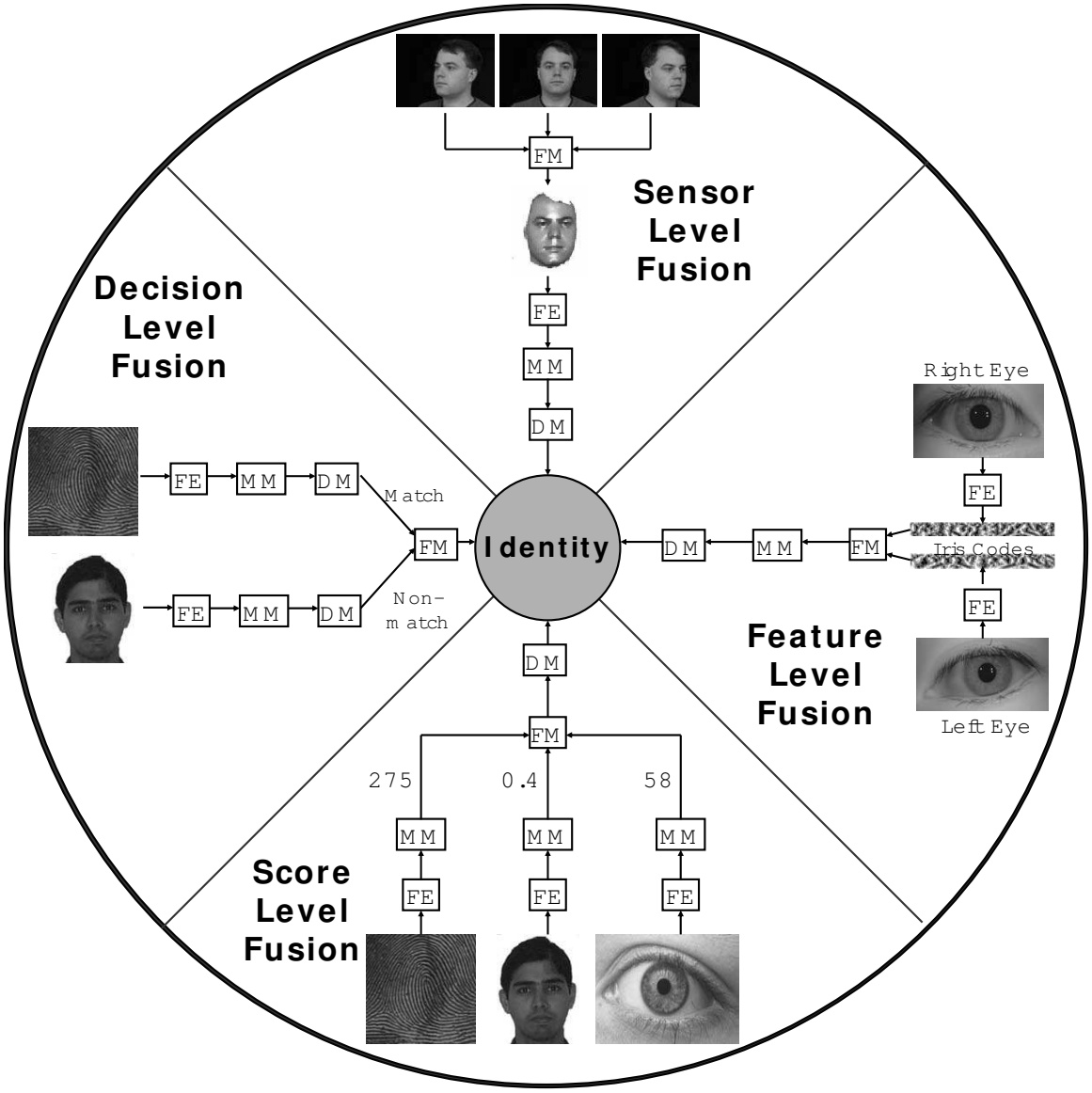



Figure 2.5: Fusion can be accomplished at various levels in a biometric system. Most multibiometric systems fuse information at the match score level or the decision level. FE: feature extraction module; MM: matching module; DM: decision-making module; FM: fusion module.

Integration at the feature level is difficult to achieve in practice due to the following reasons:

- The relationship between the feature spaces of different biometric sources may not be known. In the case where the relationship is known in advance, care needs to be taken to discard those features that are highly correlated. This requires the application of feature selection algorithms prior to classification.

- The feature sets may be incompatible. For example, the minutiae set of fingerprints and eigenface coefficients cannot be directly combined because the former is a variable length feature set whose individual values represent the attributes of a minutia point while the latter is a fixed length feature set whose individual values are scalar entities.

- Concatenating two feature vectors results in a feature vector with larger dimensionality which may lead to the 'curse of dimensionality' problem [85] where the classification accuracy actually degrades with the addition of new features due to the limited number of training samples. Although this is a well-known problem in most pattern recognition applications, it is more severe in biometric applications because of the time, effort and cost involved in collecting large amounts of biometric (training) data.

- Most commercial biometric systems do not provide access to the feature sets used in their products due to proprietary reasons.

Examples of feature level fusion schemes proposed in the literature can be found in

Chibelushi et al. [37] (voice and lip shape), Son and Lee [182] (face and iris), Kumar et al. [113] (hand geometry and palmprint) and Ross and Govindarajan [166] (face and hand geometry). Due to the constraints mentioned above, most of the attempts at feature level fusion have met with only limited success. Hence, very few researchers have studied integration at the feature level in a multibiometric system and fusion schemes at the match score and decision levels are generally preferred.

### 2.4.2 Fusion After Matching

Schemes for integration of information after the classification/matcher stage can be divided into four categories: dynamic classifier selection, fusion at the decision level, fusion at the rank level and fusion at the match score level. A *dynamic classifier selection* scheme chooses the biometric source that is most likely to give the correct decision for the specific input pattern [205]. This is also known as the winner-take-all approach and the module that performs this selection is known as an associative switch [30].

#### Score Level Fusion

Match score is a measure of the similarity between the input and template biometric feature vectors. When match scores output by different biometric matchers are consolidated in order to arrive at a final recognition decision, fusion is said to be done at the match score level. This is also known as fusion at the measurement level or confidence level. The general flow of information in a match score level fusion scheme is shown in Figure 2.6. It must be noted that the match scores generated by the indi-

vidual matchers may not be homogeneous. For example, one matcher may output a distance or dissimilarity measure (a smaller distance indicates a better match) while another may output a similarity measure (a larger similarity value indicates a better match). Furthermore, the outputs of the individual matchers need not be on the same numerical scale (range). Finally, the match scores may follow different probability distributions and may be correlated. These factors make match score level fusion a challenging problem.

### Rank Level Fusion

When the output of each biometric system is a subset of possible matches (i.e., identities) sorted in decreasing order of confidence, the fusion can be done at the *rank level*. This is relevant in an identification system where a rank may be assigned to the top matching identities. Ho et al. [79] describe three methods to combine the ranks assigned by different matchers. In the highest rank method, each possible identity is assigned the best (minimum) of all ranks computed by different systems. Ties are broken randomly to arrive at a strict ranking order and the final decision is made based on the consolidated ranks. The Borda count method uses the sum of the ranks assigned by the individual systems to a particular identity in order to calculate the fused rank. The logistic regression method is a generalization of the Borda count method where a weighted sum of the individual ranks is used. The weights are determined using logistic regression. Another technique for rank level fusion is the mixed group ranks approach [135], which attempts to find a tradeoff between the general

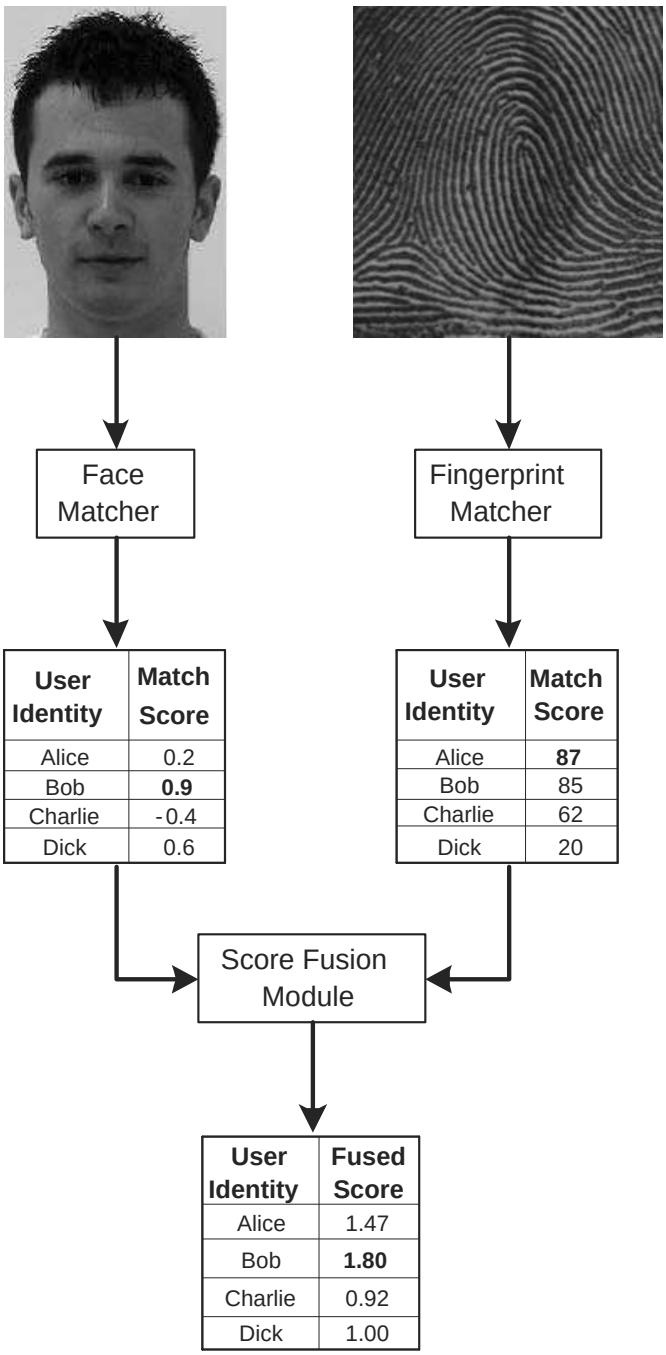



Figure 2.6: Flow of information in a match score level fusion scheme. In this example, the match scores have been combined using the sum of scores fusion rule after min-max normalization of each matcher's output. Note that the match scores generated by the face and fingerprint matchers are similarity measures. The range of match scores is assumed to be $[-1, +1]$ and $[0, 100]$ for the face and fingerprint matchers, respectively.

preference for specific matchers and the confidence in specific results (as indicated by the ranks).

#### Decision Level Fusion

In a multibiometric system, fusion is carried out at the *abstract* or *decision* level when only the decisions output by the individual biometric matchers are available. Many commercial off-the-shelf (COTS) biometric matchers provide access only to the final recognition decision. When such COTS matchers are used to build a multibiometric system, only decision level fusion is feasible. Methods proposed in the literature for decision level fusion include "AND" and "OR" rules [49], majority voting [116], weighted majority voting [114], Bayesian decision fusion [206], the Dempster-Shafer theory of evidence [206] and behavior knowledge space [82].

## 2.5 Challenges in Multibiometric System Design

While multibiometric systems offer several advantages such as better recognition accuracy, increased population coverage, greater security and flexibility, the design of a multibiometric system is not an easy task. Multibiometric system design is a challenging problem because it is very difficult to predict the optimal sources of biometric information and the optimal fusion strategy for a particular application. This difficulty arises due to the following factors.

1. **Heterogeneity of information sources**: Integration at an early stage of processing is believed to be more effective because the amount of information

available to the fusion module decreases as we move from the sensor level to the decision level. However, fusion at the sensor or feature level is not always possible due to the heterogeneity or incompatibility of the information content. For example, in a multibiometric system that uses face and fingerprint, it may not be possible to fuse either the raw images or the features extracted from them (e.g., fingerprint minutiae and eigenface coefficients).

2. **Fusion complexity**: Even when the sources of information are compatible (e.g., two impressions of the same finger, minutiae sets from two different fingers of an individual, etc.), the complexity of the fusion algorithm may nullify the advantages of fusion. For instance, fusion at the sensor or feature levels involves additional processing complexities such as registration and design of new algorithms to match the fused data. Further, the raw data from the sensor and the extracted feature sets are usually corrupted by various types of noise (e.g., background clutter in a face image, spurious minutiae in a fingerprint minutiae set, etc.). Hence, fusion at the sensor and feature level may not lead to any performance improvement.

3. **Varied discriminative ability**: The amount of discriminatory information provided by each biometric source can be quite different. Consider a multibiometric system with two matchers A and B, where the matcher A has very high accuracy compared to matcher B. If a simple fusion rule that assigns equal weights to the information from the two matchers is employed, the accuracy of the multibiometric system is likely to be lower than the accuracy of the individ-

ual matcher A. Furthermore, some multibiometric systems utilize soft biometric traits like gender, ethnicity, height, etc., which have significantly lower discriminatory information content compared to traditional biometric identifiers such as fingerprint, face and iris. Hence, it is essential to estimate the amount of discriminatory information in each source and assign appropriate weights to the different sources based on their information content.

4. **Correlation between sources**: In many multibiometric systems, the different biometric sources may not be statistically independent. Examples of multibiometric systems in which different information sources are correlated include (i) systems using physically related traits (e.g., speech and lip movement of a user), (ii) multiple matchers operating on the same biometric data or feature representation (e.g., two different face matchers that operate on the same raw face image) and (iii) multiple samples of the same biometric trait (e.g., two impressions of a person's right index finger). In general, fusion of independent evidences can be expected to provide a larger improvement in accuracy compared to fusion of correlated sources. But the impact of correlation among the biometric sources on the fusion performance is not completely known.

Apart from the above four factors, the conflicting performance requirements of an application also contribute to the difficulty of the fusion problem. A typical example is an identification system where both the accuracy and throughput requirements need to be satisfied. While utilizing more sources of evidence increases the accuracy, it may reduce the throughput of the system and it is hard to find the optimal tradeoff

between the two. Due to these reasons, information fusion in biometrics is still an active area of research despite the fact that information fusion has been well studied in the wider pattern recognition context.

## 2.6 Summary

Multibiometric system design depends on various factors such as sources of information, acquisition and processing architecture, level of information fusion and fusion methodology. There has been a proliferation of work exploring the fusion of a variety of biometric sources and discussing different fusion techniques. Tables 2.1, 2.2, 2.3, and 2.4 summarize some of the representative work in the multibiometrics literature and these tables have been categorized based on the sources of information used.

From these tables, it is quite apparent that fusion at the match score level has received the maximum attention from the biometrics community. However, most of the proposed score level fusion schemes involve ad-hoc techniques for normalizing the match scores and assigning optimal weights to different matchers. Hence, one of the goals of this dissertation is to develop a principled statistical framework for match score fusion in multibiometric systems. Score fusion in a multibiometric verification system can be formulated as a two-class classification problem and a significant number of training samples are usually available for both the genuine and impostor classes. On the other hand, fusion in multibiometric identification systems is typically characterized by (i) a large number of classes (identities), (ii) frequent change in the number of classes during system operation due to addition/deletion

of users and (iii) insufficient number of training samples for the individual classes (often, only one score per matcher is available available for each user). Due to these reasons, we consider the fusion strategies for verification and identification systems separately in this dissertation. In chapter 3, we present a likelihood ratio based fusion framework for multibiometric verification systems. The fusion framework for multibiometric identification is presented in chapter 4. Furthermore, while template security has been receiving substantial attention, the issue of multibiometric template security has not been adequately addressed in the literature. Therefore, in chapter 5 we develop techniques that can protect multibiometric templates as a single entity.

Table 2.1: Examples of multi-sensor systems.

| **Sensors Fused** | **Authors** | **Level of Fusion** | **Fusion Methodology** |
|---|---|---|---|
| Optical and capacitive fingerprint sensors | [130] | Match score | Sum and product rules; logistic regression |
| 2D camera and range scanner for face | [26] | Match score | Weighted sum and product rules |
| | [124] | Match score | Weighted sum rule; hierarchical matching |
| 2D camera and IR camera for face | [181] | Match score | Weighted sum rule |
| | [31] | Match score; rank | Sum rule; logistic regression |
| 2D camera, range scanner and IR camera for face | [27] | Match score | Weighted sum rule |
| Red, Green, Blue channels for face | [109] | Match score | Sum and min rules |
| | [166] | Feature; match score | Feature selection and concatenation; sum rule |

Table 2.2: Examples of multi-algorithm systems.

| **Representations and/or Matchers Fused** | **Authors** | **Level of Fusion** | **Fusion Methodology** |
|---|---|---|---|
| Fingerprint (minutiae and texture features) | [132, 155, 168] | Match score | Likelihood ratio, weighted sum rule, sum and product rules, perceptron |
| Face (PCA, LDA, ICA) | [125, 131, 166] | Feature, match score | Sum and max rules, nearest neighbor, RBF network, feature selection and concatenation |
| Face (LDA, PM, HST) | [45] | Match score | Sum, product, min, max and median rules; quadratic Bayes; Parzen; weighted sum rule |
| Face (global and local features) | [59] | Feature | ANFIS (Adaptive Neuro-Fuzzy Inference System); SVM |
| Face (two different sets of PCA-based features) | [208] | Feature | Feature concatenation; two sets of features form the real and imaginary parts of the concatenated feature vector in the complex plane |
| Signature (global and local features) | [64, 71] | Match score | Sum and max rules, SVM |
| Hand (geometry and texture features) | [112] | Feature; match score | Feature concatenation; sum rule |
| Voice (SVM and GMM) | [20] | Match score | Weighted sum rule; perceptron |
| Voice (multi-level features) | [162] | Match score | Perceptron |
| Voice (spectral features, utterance verification) | [165] | Match score | Sum, product, min, max and median rules; neural network |
| Voice (LPCC, MFCC, ARCSIN, FMT features) | [107] | Feature; match score; decision | Feature concatenation; sum rule; majority voting |
| Voice (MFCC, CMS, MACV features) | [172] | Feature; match score | Feature concatenation; weighted sum rule |
| Palmprint (Gabor, line, appearance-based) | [113] | Match score; decision | Sum rule (for Gabor and line features) followed by product rule; SVM; neural network; AND rule |
| Palmprint (geometry, texture, fuzzy "interest" line) | [211] | Decision | Hierarchical (serial) matching |

Table 2.3: Examples of multi-sample and multi-instance systems.

| **Modality** | **Authors** | **Level of Fusion** | **Fusion Methodology** |
|---|---|---|---|
| Fingerprint (10 fingers) | [204] | Match score | No details are available |
| Fingerprint (2 fingers) | [72] | Match score | Sum rule |
| Fingerprint (2 impressions, 2 fingers) | [155] | Match score | Likelihood ratio computed from non-parametric joint density estimates |
| Fingerprint (2 impressions) | [95] | Sensor; feature | Mosaicing of templates at the image level; mosaicing of minutiae sets |
| | [140] | Feature | Mosaicing of minutiae sets |
| Face (sequence of images from video) | [213] | Match score | Temporal integration |
| | [121] | Match score | Temporal integration through construction of identity surfaces |
| Voice (multiple utterances) | [36] | Match score | Zero sum fusion after sorting of scores |

Table 2.4: Examples of multimodal systems.

| **Modalities Fused** | **Authors** | **Level of Fusion** | **Fusion Methodology** |
|---|---|---|---|
| Face and voice | [15] | Match score; rank | Geometric weighted average; HyperBF |
| | [108] | Match score | Sum, product, min, max and median rules |
| | [6] | Match score | SVM; multilayer perceptron; C4.5 decision tree; Fisher's linear discriminant; Bayesian classifier |
| | [10] | Match score | Statistical model based on Bayesian theory |
| Face, voice and lip movement | [69] | Match score; decision | Weighted sum rule; majority voting |
| Face and fingerprint | [80] | Match score | Product rule |
| | [180] | Match score | Sum rule, Weighted sum rule |
| Face, fingerprint and hand geometry | [167] | Match score | Sum rule; decision trees; linear discriminant function |
| Face, fingerprint and voice | [88] | Match score | Likelihood ratio |
| Face and iris | [203] | Match score | Sum rule; weighted sum rule; Fisher's linear discriminant; neural network |
| Face and gait | [178] | Match score | Sum rule |
| | [104] | Match score | Sum and product rules |
| Face and ear | [25] | Sensor | Concatenation of raw images |
| Face and palmprint | [60] | Feature | Feature concatenation |
| Fingerprint, hand geometry and voice | [190] | Match score | Weighted sum rule |
| Fingerprint and hand geometry | [191] | Match score | Reduced multivariate polynomial model |
| Fingerprint and voice | [192] | Match score | Functional link network |
| Fingerprint and signature | [65] | Match score | SVM in which quality measures are incorporated |
| Voice and signature | [111] | Match score | Weighted sum rule |

# Chapter 3

# Multibiometric Verification

While fusion in a multibiometric verification system can be performed at the sensor, feature, match score and decision levels, score level fusion is generally preferred because it offers the best trade-off in terms of the information content and the ease in fusion. One of the challenges in combining match scores is that scores from different matchers are typically not homogeneous. Consider the scores provided by the two face matchers in the NIST-Face database [151]. The scores from the first face matcher are in the range $[-1, 1]$, whereas scores from the second face matcher are in the range $[0, 100]$ (see Figure 3.1). The match scores of different matchers (i) can be either distance or similarity measures, (ii) may follow different probability distributions and (iii) matcher accuracies may be quite different. For example, in the case of the MSU-Multimodal database [90], the fingerprint matcher outputs similarity scores whereas the face matcher outputs distance scores; the score distributions for these two modalities are quite different (see Figure 3.3) and the fingerprint matcher is more accurate than the face matcher. Biometric matchers may also be correlated as shown

in Figure 3.1; the correlation coefficient[1] for the genuine and impostor scores of the two face matchers in Figure 3.1 are 0.7 and 0.3, respectively.

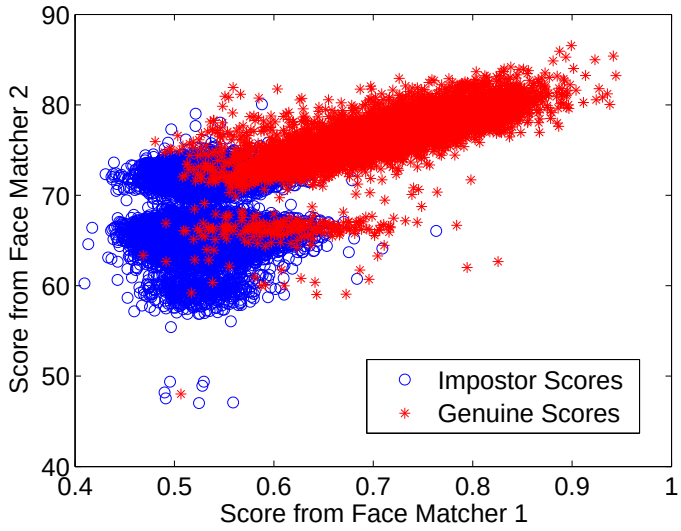



Figure 3.1: Non-homogeneity in the match scores provided by the two face matchers in the NIST-Face database. Note that about 0.2% of the scores output by matcher 1 are discrete scores with value -1, which are not shown in this plot.

Score fusion techniques can be divided into the following three categories.

- *Transformation-based score fusion:* The match scores are first normalized (transformed) to a common domain and then combined using product, sum, max or min rules [108]. Choice of the normalization scheme and combination weights is data-dependent and requires extensive empirical evaluation [90, 167, 180, 190].

- *Classifier-based score fusion:* Scores from multiple matchers are treated as a feature vector and a classifier is constructed to discriminate genuine and impostor

[1] In this dissertation, we estimate correlation using the Pearson's product-moment correlation coefficient, which measures the strength and direction of linear relationship between two random variables [164]. The correlation between two matchers is defined as the correlation between the scores of the two matchers.

scores [15,66,127]. When biometric score fusion is considered as a classification problem, the following issues pose challenges. (i) Unbalanced training set: The number of genuine match scores available for training is $O(N)$, but the number of impostor scores is $O(N^2)$, where $N$ is the number of users in the database. (ii) Cost of misclassification: Depending on the biometric application, the cost of accepting an impostor may be very different from the cost of rejecting a genuine user. For example, a biometric system deployed in a security application typically is required to have a false accept rate (FAR) of less than 0.1%. Therefore, the fusion strategy needs to minimize the false reject rate (FRR) at the specified FAR values rather than minimizing the total error rate (sum of FAR and FRR) [155]. (iii) Choice of classifier: Given a variety of admissible classifiers, selecting and training a classifier that gives the optimal performance (minimum FRR at a specified FAR) on a given data set is not easy.

- *Density-based score fusion:* This approach is based on the likelihood ratio test and it requires explicit estimation of genuine and impostor match score densities [74,155]. The density based approach has the advantage that it directly achieves optimal performance at any desired operating point (FAR), provided the score densities can be estimated accurately. In fact, a comparison of eight biometric fusion techniques conducted by NIST [195] with data from 187,000 subjects concluded that "Product of Likelihood Ratios was consistently most accurate, but most complex to implement" and "complexity in this implementation is in the modeling of distributions, rather than fusion per se". The statement

in [195] about the complexity of density estimation was based on the use of kernel density estimator (KDE). The selection of kernel bandwidth and density estimation at the tails proved to be the most complex steps in estimating the score densities using KDE in [195].

Among the three approaches, density based fusion is a more principled approach because it achieves optimal fusion performance if the score densities are estimated accurately. Hence, we follow the density-based score fusion approach in this thesis. We investigate two different techniques for accurately estimating the genuine and impostor match score densities, namely, the Gaussian mixture model (GMM) and the non-parametric kernel density estimator (KDE). We show that (i) GMM is quite effective in modeling the genuine and impostor score densities and is simpler to implement than KDE, (ii) fusion based on the resulting density estimates achieves consistently high performance on three multibiometric databases involving face, fingerprint, iris, and speech modalities and (iii) biometric sample quality can be easily incorporated in the likelihood ratio based fusion framework.

## 3.1 Likelihood Ratio Test

Let $S$ be a random variable denoting the match score provided by a matcher. Let the distribution function for the genuine scores be denoted as $F_{gen}(s)$ (i.e., $P(S \leq s|S \text{ is genuine}) = F_{gen}(s)$) with the corresponding density function $f_{gen}(s)$. Similarly, let the distribution function for the impostor scores be denoted as $F_{imp}(s)$ with the corresponding density function $f_{imp}(s)$. Suppose we need to decide between

the genuine and impostor classes (to verify a claimed identity) based on the observed match score $s$. Let $\Psi$ be a statistical test for testing the null hypothesis $H_0$: *score $S$ corresponds to an impostor* against the alternative hypothesis $H_1$: *score $S$ corresponds to a genuine user*. Let $\Psi(s) = i$ imply that we decide in favor of $H_i$, where $i = 0, 1$. The probability of rejecting $H_0$ when $H_0$ is true is known as the *false accept rate* (also referred to as the *size* or *level* of the test). The probability of correctly rejecting $H_0$ when $H_1$ is true is known as the *genuine accept rate* (also referred to as the *power* of the test). The Neyman-Pearson theorem [118] states that

1. For testing $H_0$ against $H_1$, there exists a test $\Psi$ and a constant $\eta$ such that

$$P(\Psi(S) = 1|H_0) = \alpha \tag{3.1}$$

and

$$\Psi(s) = \begin{cases} 1, & \text{when } \frac{f_{gen}(s)}{f_{imp}(s)} > \eta, \\ 0, & \text{when } \frac{f_{gen}(s)}{f_{imp}(s)} < \eta. \end{cases} \tag{3.2}$$

When $f_{gen}(s)/f_{imp}(s)$ is equal to $\eta$, $\Psi(s)$ is zero with probability $\gamma$ and one with probability $1-\gamma$. Here, $\gamma$ is chosen such that the level of the test is exactly equal to $\alpha$.

2. If a test satisfies equations (3.1) and (3.2) for some $\eta$, then it is the most powerful test for testing $H_0$ against $H_1$ at level $\alpha$.

According to the Neyman-Pearson theorem, given the false accept rate (FAR) $\alpha$, the *optimal* test for deciding whether a match score $S$ corresponds to a genuine user or an impostor is the likelihood ratio test given by equation (3.2). For a fixed FAR, we can select a threshold $\eta$ such that the likelihood ratio test maximizes the genuine accept rate (GAR) and there does not exist any other decision rule with a higher GAR. However, this optimality of the likelihood ratio test is guaranteed only when the underlying densities are known. In practice, we only have a finite set of genuine and impostor match scores, so we need to reliably estimate the densities $f_{gen}(s)$ and $f_{imp}(s)$ before applying the likelihood ratio test.

## 3.2 Estimation of Match Score Densities

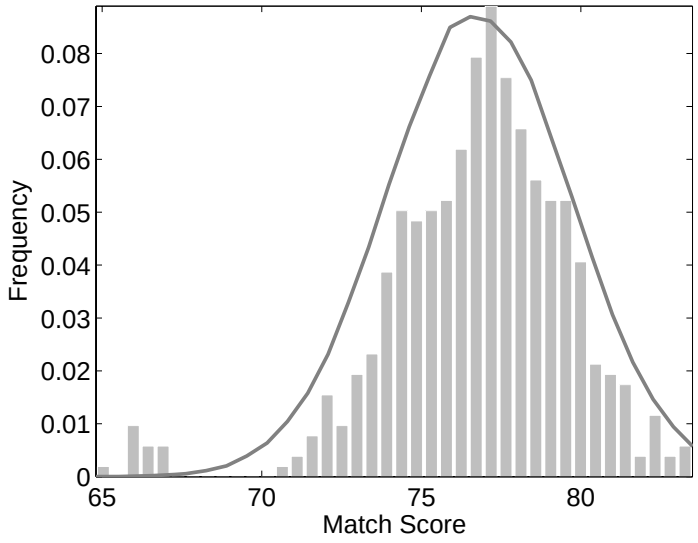
Density estimation techniques can be classified as parametric or non-parametric [179]. In parametric density estimation, the form of the density function (e.g., Gaussian) is assumed to be known and only the parameters of this density function (e.g., mean and standard deviation) are estimated from the training data. Non-parametric techniques (e.g., density histogram and kernel density estimator) do not assume any standard form for the density function and are essentially data-driven. A mixture of densities whose functional forms are known (e.g., mixture of Gaussians) can also be used for density estimation. This mixture method can be categorized as either parametric or semi-parametric depending on whether the number of mixture components is fixed *a priori* or is allowed to vary based on the observed data [67].

In the context of biometric systems, it is very difficult to choose a specific para-

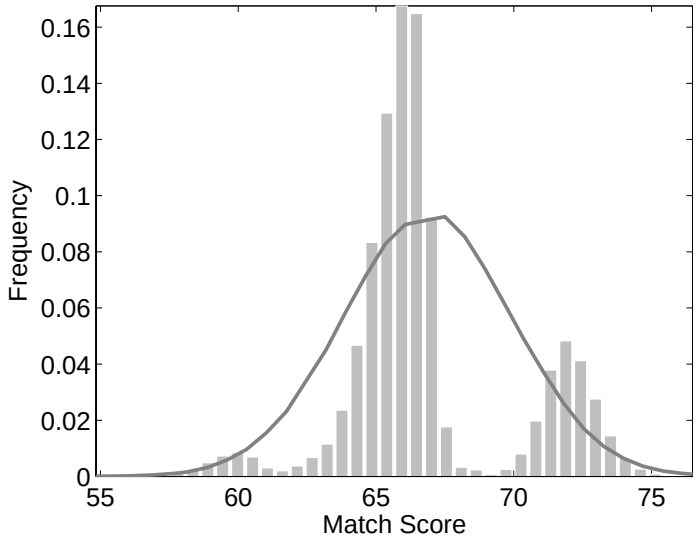
metric form for the density of genuine and impostor match scores. It is well known that the Gaussian density is usually not appropriate for genuine and impostor match scores because the score distributions generally have a large tail and may have more than one mode (see Figure 3.2). The simplest non-parametric density estimator is the histogram method, which has the following limitations [202]: (i) it is sensitive to the placement of the bin-edges, (ii) it estimates the density by a step function and (iii) the asymptotic rate of convergence[2] of the histogram is lower than that of other density estimators. Due to the above reasons, we do not use histograms for estimating the score densities.

Griffin [74] used the following non-parametric approach to estimate the match score densities. The distribution functions $F_{gen}(s)$ and $F_{imp}(s)$ are approximated using polynomials whose coefficients are obtained empirically from the receiver operating characteristic (ROC) curve of the biometric matcher. The marginal densities $f_{gen}(s)$ and $f_{imp}(s)$ are then obtained by differentiating the corresponding distribution functions. Although this method is relatively simple, the main limitation is that the choice of polynomial degree to be used for approximating the distribution functions is arbitrary. Further, there is no guarantee that the estimated densities will converge to the true underlying densities. To overcome these limitations, Prabhakar and Jain [155] used kernel density estimators (also known as the Parzen window method [57]) for estimating the score densities.

[2]The asymptotic rate of convergence of a density estimator is defined as the rate at which the integrated mean squared error between the true and estimated densities approaches zero as the number of samples available for density estimation tends to infinity.




(a)




(b)

Figure 3.2: Histograms of match scores and the corresponding Gaussian density estimates for the Face-G matcher in the NIST BSSR1 database. (a) Genuine and (b) Impostor. Note that the Gaussian density does not account well for the tail in the genuine score distribution and the multiple modes in the impostor score distribution.

### 3.2.1 Kernel Density Estimation

In practice, many biometric matchers apply thresholds at various stages in the matching process. When the required threshold conditions are not met, pre-specified match scores are output by the matcher. For example, a fingerprint matcher may output a specific score value (say $s_1$) if the orientation field of the input fingerprint does not match well with the template; the same matcher may provide a different score value (say $s_2$) if the number of minutia points in the input fingerprint is less than a threshold. This leads to discrete components in the match score distribution that cannot be modeled accurately using a continuous density function. Hence, we propose a modified kernel density estimator [48] in which the marginal density is modeled as a mixture of continuous and discrete components (referred to as generalized density) and the joint density is estimated using copula functions.

#### Generalized Marginal Density

A generic score value $s_0$ is said to be discrete if $P(S = s_0) > 0$. In such a situation, $F$ cannot be represented by a density function in the neighborhood of $s_0$ (since this would imply that $P(S = s_0) = 0$). Hence, our approach consists of first detecting discrete components in the genuine and impostor match score distributions, and then modeling the observed distribution of match scores as a mixture of discrete and continuous components.

Given a set of match scores, $\mathcal{S}$, we first identity if there are any discrete components in it, namely, score values $s_0$ with $P(S = s_0) \geq T$, where $T$ is a threshold;

$0 \leq T \leq 1$. The value of $T$ can be determined using the algorithm described in the Appendix B.1. We estimate the probability $P(S = s_0)$ by $\frac{N(s_0)}{N}$, where $N(s_0)$ is the number of observations in $\mathcal{S}$ that equal $s_0$ and $N$ is the total number of observations. The collection of all discrete components for a match score distribution is denoted by

$$\mathcal{D} \equiv \{\, s_0 \,:\, \frac{N(s_0)}{N} \geq T\}. \tag{3.3}$$

The discrete components constitute a proportion $p_D \equiv \sum_{s_0 \in \mathcal{D}} \frac{N(s_0)}{N}$ of the complete set of match scores, $\mathcal{S}$. We obtain the subset $\mathcal{C}$, $\mathcal{C} \subseteq \mathcal{S}$, by removing all the discrete components from $\mathcal{S}$, $\mathcal{C} = \mathcal{S} - \mathcal{D}$. The scores in $\mathcal{C}$ constitute a proportion $p_C \equiv (1 - p_D)$ of $\mathcal{S}$, and they are used to estimate the continuous component of the density ($f_C(s)$). The continuous component of match score density is estimated using a kernel density estimate of $f_C(s)$, which is given by

$$\hat{f}_C(s) = \frac{1}{hN_C} \sum_{x \in \mathcal{C}} \mathcal{K}\left(\frac{s - x}{h}\right), \tag{3.4}$$

where $\mathcal{K}$ is a function satisfying $\int_{-\infty}^{\infty} \mathcal{K}(s)ds = 1$, called the *kernel*, $h$ is a positive number, called the *bandwidth* of the kernel and $N_C \equiv N\, p_C$. Usually $\mathcal{K}$ is chosen to be a unimodal probability density function symmetric about zero. We use the Gaussian kernel ($\mathcal{K}(s) = \phi(s)$, where $\phi(s)$ is the standard normal density) for density estimation.

The choice of kernel bandwidth is a critical factor in kernel density estimation. In [155], a simple heuristic was used to estimate the bandwidth of the kernel (set to $0.01\hat{\sigma}$, where $\hat{\sigma}$ is the standard deviation of the observed match scores). However, the

above heuristic is not always optimal and does not provide accurate density estimates on a variety of multibiometric databases. Hence, we use an automatic bandwidth estimator known as "solve-the-equation" bandwidth selector [202] to obtain the optimal bandwidth. The "solve-the-equation" bandwidth estimator has been shown to give very good density estimates for a large class of underlying functions. This bandwidth estimator minimizes a mean square error criterion asymptotically. In other words, the density estimate obtained from the "solve-the-equation" bandwidth estimator preserves most of the characteristics (e.g., peaks and tails) of the distribution of match scores without over-smoothing, thus, achieving a good compromise between the bias and the variance of the density estimate (see Figure 3.3).

The generalized density (a mixture of discrete and continuous components) is defined as

$$\hat{f}(s) = p_C \, \hat{f}_C(s) + \sum_{s_0 \in \mathcal{D}} \frac{N(s_0)}{N} \cdot I\{s = s_0\}, \tag{3.5}$$

where $I\{x = s_0\} = 1$ if $s = s_0$, and 0, otherwise. The distribution function corresponding to the generalized density estimate is defined as

$$\hat{F}(s) = p_C \int_{-\infty}^{s} \hat{f}_C(u)\, du + \sum_{s_0 \in \mathcal{D},\, s_0 \leq s} \frac{N(s_0)}{N}. \tag{3.6}$$

The above approach for estimating the generalized density can be applied to the genuine and impostor match scores from different matchers. For a multibiometric system with $K$ matchers, we denote the $k^{th}$ generalized marginal density estimated from the genuine scores as $\hat{f}_{gen,k}(s)$, $k = 1, 2, \ldots, K$. The corresponding estimates

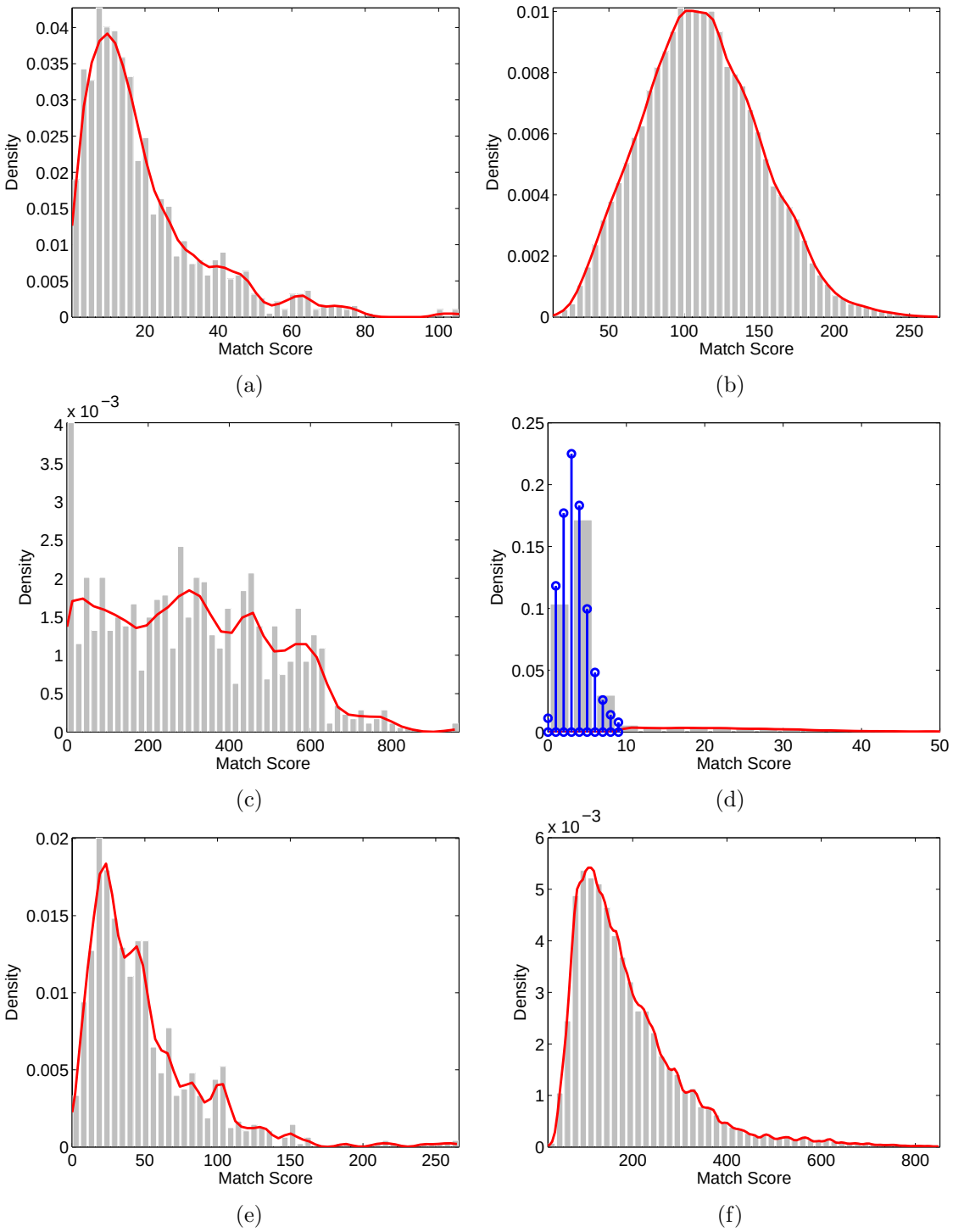


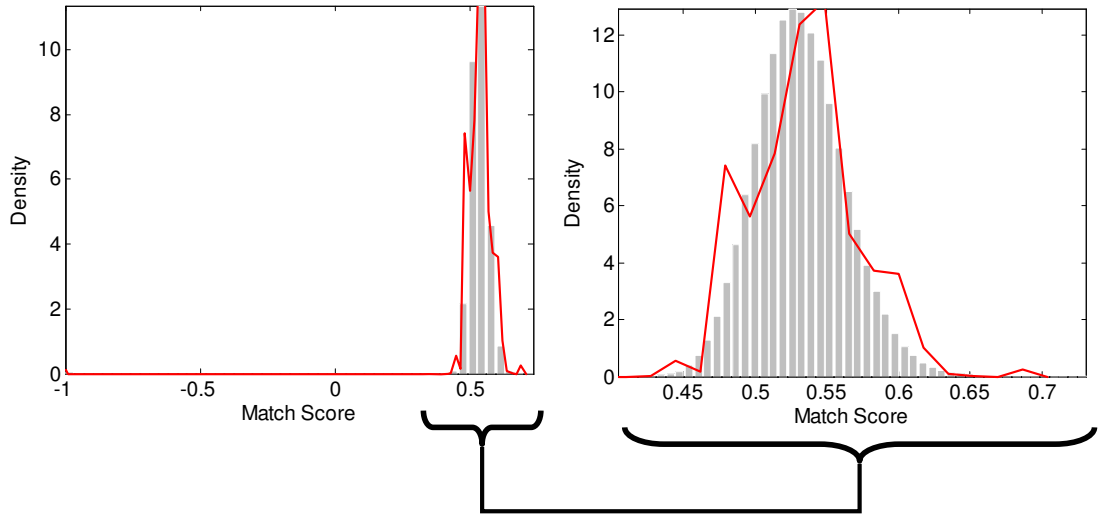

Figure 3.3: Histograms of match scores and the corresponding generalized density estimates for MSU-Multimodal database. (a) and (b) Genuine and impostor match scores for face modality. (c) and (d) Genuine and impostor match scores for fingerprint modality. (e) and (f) Genuine and impostor match scores for hand geometry modality. The solid line above the histogram bins is the density estimated using the kernel density estimator, and the spikes in (d) correspond to the discrete components.

based on the impostor scores are denoted by $\hat{f}_{imp,k}(s),\ k = 1, 2, \ldots, K$. Figure 3.3 gives the plots of $\hat{f}_{gen,k}(s)$ and $\hat{f}_{imp,k}(s),\ k = 1, 2, 3$ for the distribution of observed genuine and impostor match scores in the MSU-Multimodal database (see Appendix for a description of this database). Figure 3.3 also gives the histograms of the genuine and impostor match scores for the three modalities, namely, face, fingerprint and hand-geometry. Discrete components were detected only in the case of impostor match scores of the fingerprint modality; see the "spikes" in Figure 3.3(d) that represent the detected discrete components for $T = 0.008$ in equation (3.3).

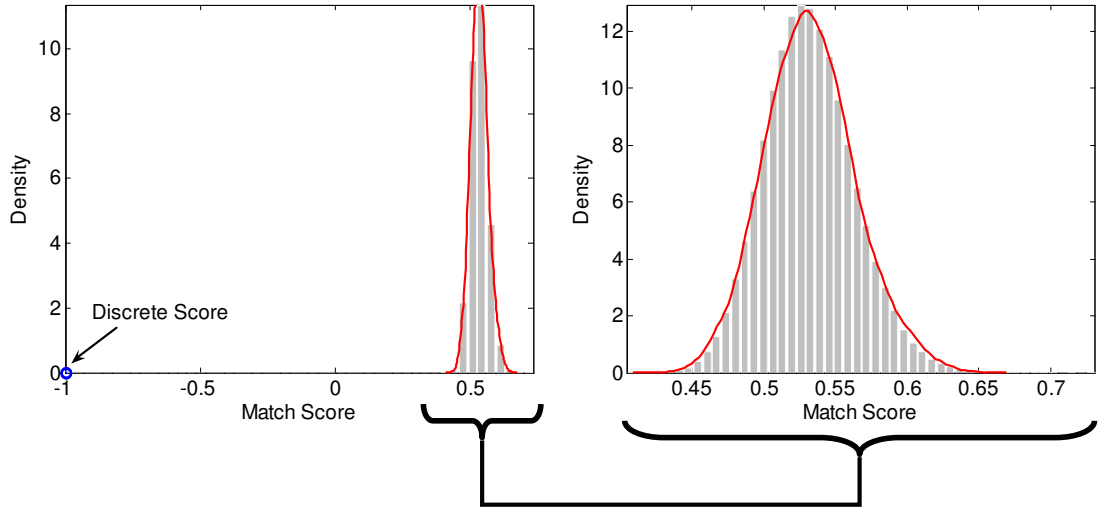
A comparison between the continuous and generalized density estimates for impostor match scores provided by the first face matcher in the NIST-Face database is shown in Figure 3.4. This matcher can output a discrete match score with value $-1$. Figure 3.4(a) shows the continuous density estimate over the entire range of scores ($[-1, 1]$) and the same estimate only in the range $[0.4, 0.7]$ that covers a majority of the scores. The scores with value $-1$ affect the kernel bandwidth significantly ($h = 0.00001$ when the scores with value $-1$ are present, while $h = 0.0027$ when they are removed). As a result, the continuous density estimates of the impostor scores are not accurate in the range $[0.4, 0.7]$. On the other hand, the generalized density estimates shown in Figure 3.4(b) are very accurate in modeling the match scores.

### Generalized Multivariate Density Using Copula Models

The methodology described in section 3.2.1 provides only the marginal genuine and impostor score densities for each of the $K$ matchers. When the matchers are assumed to be mutually independent, the joint (multivariate) density of the $K$ match scores can




(a)




(b)

Figure 3.4: Comparison of continuous and generalized density estimates for impostor match scores provided by the first face matcher in the NIST-Face database. (a) Continuous density estimates in the entire score range $[-1, 1]$ and only in the range $[0.4, 0.7]$. (b) Generalized density estimates ($T = 0.002$) in the entire score range $[-1, 1]$ and only in the range $[0.4, 0.7]$.

be estimated as the product of the marginal densities. However, if the matchers are correlated, it may be important to model the dependence between them. When the marginal distributions are continuous, the joint density can be directly estimated using multidimensional kernels. Since the marginal distribution of match scores contains discrete components, we use copula functions [146] to estimate the multivariate distribution. The copula-based joint density estimation is semi-parametric because the marginals are non-parametric and the copula function that combines the marginals to get the joint density is parametric.

Let $H_1, H_2, \ldots, H_K$ be $K$ continuous distribution functions and $H$ be a $K$-dimensional distribution function with the $k^{th}$ marginal given by $H_k$, $k = 1, 2, \ldots, K$. According to Sklar's theorem [146], there exists a unique function $C(u_1, u_2, \ldots, u_K)$ from $[0, 1]^K$ to $[0, 1]$ satisfying

$$H(s_1, s_2, \ldots, s_K) = C(H_1(s_1), H_2(s_2), \ldots, H_K(s_K)), \tag{3.7}$$

where $s_1, \ldots, s_K$ are $K$ real numbers. The function $C$ is known as a $K$-copula function that "couples" the univariate distributions $H_1, H_2 \ldots, H_K$ to obtain the $K$-variate distribution $H$.

We use the family of Gaussian copula functions [34] to model the joint distributions of match scores[3]. These functions incorporate the second-order dependence among the $K$ matchers using a $K \times K$ correlation matrix $R$. The $K$-dimensional Gaussian copula function is given by

[3]The Gaussian copula function does not assume that the joint or marginal match score distributions are Gaussian.

$$C_R^K(u_1, u_2, \ldots, u_K) = \Phi_R^K(\Phi^{-1}(u_1), \Phi^{-1}(u_2), \ldots, \Phi^{-1}(u_K)), \tag{3.8}$$

where each $u_k \in [0, 1]$ for $k = 1, 2, \ldots, K$, $R$ is the correlation matrix, $\Phi(\cdot)$ is the distribution function of the standard normal, $\Phi^{-1}(\cdot)$ is its inverse, and $\Phi_R^K$ is the $K$-dimensional distribution function of a random vector $\mathcal{Z} = (Z_1, Z_2, \ldots, Z_K)^T$ with component means and variances given by 0 and 1, respectively. The density of $C_R^K$, denoted by $c_R^K$, is defined as

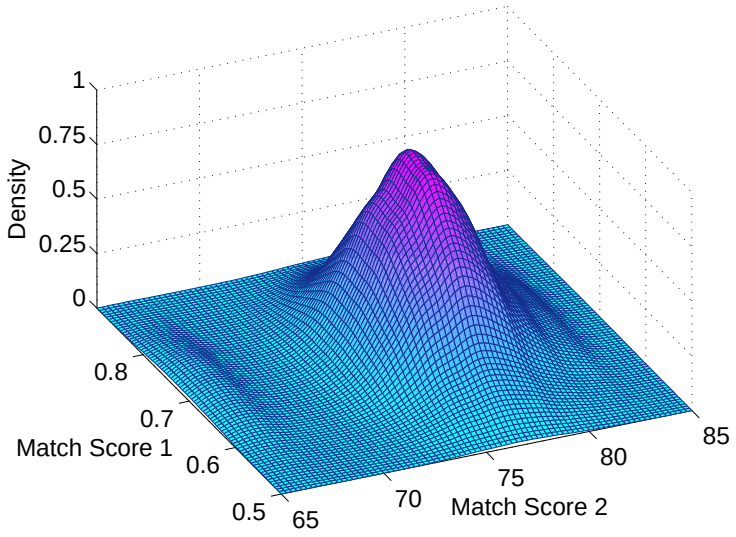
$$c_R^K(u_1, u_2, \ldots, u_K) \equiv \frac{\partial C_R^K(u_1, u_2, \ldots, u_K)}{\partial u_1 \partial u_2 \ldots \partial u_K} = \frac{\phi_R^K(\Phi^{-1}(u_1), \Phi^{-1}(u_2), \ldots, \Phi^{-1}(u_K))}{\prod_{k=1}^K \phi(\Phi^{-1}(u_k))}, \tag{3.9}$$

where $\phi_R^K(s_1, s_2, \ldots, s_K)$ is the density function of the $K$-variate normal distribution with mean 0 and covariance matrix $R$ (since the variance of each component of $\mathcal{Z}$ is 1, the covariance matrix is the same as the correlation matrix $R$), and $\phi(x)$ is the standard normal density.

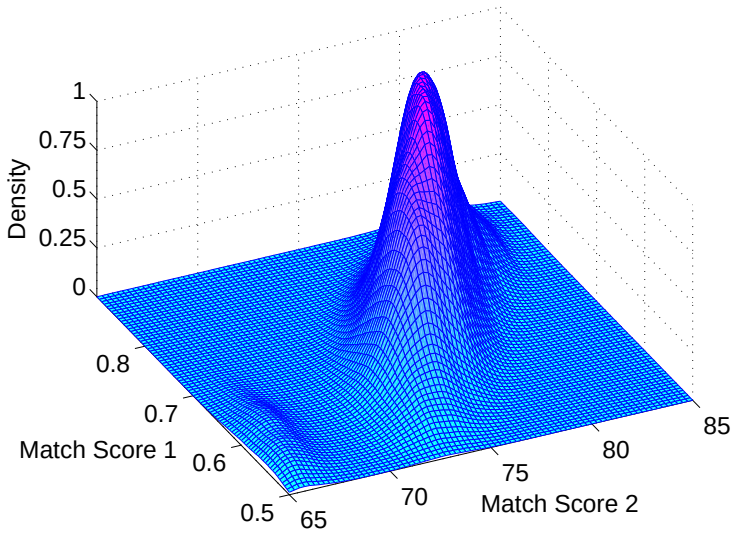
The $(m, n)$-th entry of $R$, $\rho_{mn}$, measures the degree of correlation between the $m^{th}$ and $n^{th}$ matchers for $m, n = 1, 2, \ldots, K$. Since the $K \times K$ correlation matrix $R$ is unknown, we estimate it using the Pearson's product-moment correlation of normal quantiles [164] corresponding to the given match scores from the $K$ matchers. This method assumes that the $K$ match scores come from the multivariate distribution $H$ with *continuous* marginals, $H_1, H_2, \ldots, H_K$. However, the marginals associated with the genuine and impostor distributions of the $K$ matchers may have discrete compo-

nents. Therefore, the generalized distributions are first "converted" into continuous distributions. This is achieved by perturbing each discrete component of $\hat{F}_{gen,k}(s)$ and $\hat{F}_{imp,k}(s)$ through the addition of a Gaussian noise process with mean 0 and standard deviation $\sigma = 0.0001$. Note that the discrete scores are perturbed only when estimating $R$, and not during the estimation of the marginal distributions $\hat{F}_{gen,k}(s)$ and $\hat{F}_{imp,k}(s)$. Hence, the multivariate density obtained by using the copula function is still a generalized (mixture of discrete and continuous) density.

We model the joint distribution function of genuine match scores for $K$ matchers, $F^K_{gen}$, as shown in equations (3.7) and (3.8) for some correlation matrix $R_{gen}$. For the genuine case, the $k^{th}$ marginal will be estimated by $\hat{F}_{gen,k}(s)$ for $k = 1, 2, \ldots, K$. The joint distribution function of the impostor match scores, $F^K_{imp}$, is of the same form as $F^K_{gen}$, but with a correlation matrix $R_{imp}$. In the impostor case, the $k^{th}$ marginal is estimated by $\hat{F}_{imp,k}(s)$ for $k = 1, 2, \ldots, K$. Figure 3.5 shows the joint density estimates of the genuine match scores output by the two matchers in the NIST-Face database when they are estimated using (i) product of the marginals (under the assumption of statistical independence) and (ii) copula functions. We can observe that the joint density estimated using copula functions is able to capture the correlation between the two face matchers (see Figure 3.5(b)) and hence, is a better estimate of the underlying genuine match score density.




(a)




(b)

Figure 3.5: Joint density of the genuine match scores output by the two matchers in the NIST-Face database estimated using (a) product of marginal densities and (b) copula functions. The density estimate in (b) captures the correlation between the matchers.

### 3.2.2 GMM-based Density Estimation

Although the modified kernel density estimation approach resulted in good fusion performance [48], it is not clear whether our heuristic used for detecting the discrete components and the use of a parametric copula function to estimate the joint density are optimal. To avoid these issues, we employ a well-known technique based on Gaussian mixture models (GMM) for density estimation [142]. Note that Gaussian mixture models can be used to estimate arbitrary densities and the theoretical results in [119, 157] show that the density estimates obtained using finite mixture models indeed converge to the true density.

Let $\boldsymbol{S} = [S_1, S_2, \cdots, S_K]$ be the random vector corresponding to the match scores of $K$ different biometric matchers, where $S_k$ is the random variable representing the match score provided by the $k^{th}$ matcher, $k = 1, 2, \cdots, K$. Let $f_{gen}(\boldsymbol{s})$ and $f_{imp}(\boldsymbol{s})$ be the conditional joint density of the $K$ match scores given the genuine and impostor classes, respectively, where $\boldsymbol{s} = [s_1, s_2, \cdots, s_K]$. Let $\phi^K(\boldsymbol{s}; \boldsymbol{\mu}, \Sigma)$ be the K-variate Gaussian density with mean vector $\boldsymbol{\mu}$ and covariance $\Sigma$, i.e.,

$$\phi^K(\boldsymbol{s}; \boldsymbol{\mu}, \Sigma) = (2\pi)^{-K/2}|\Sigma|^{-1/2} \exp\left(-\frac{1}{2}(\boldsymbol{s}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{s}-\boldsymbol{\mu})\right). \tag{3.10}$$

The estimates of $f_{gen}(\boldsymbol{s})$ and $f_{imp}(\boldsymbol{s})$ are obtained as a mixture of Gaussians as follows.

$$\hat{f}_{gen}(\boldsymbol{s}) = \sum_{j=1}^{M_{gen}} p_{gen,j}\phi^K\left(\boldsymbol{s}; \boldsymbol{\mu}_{gen,j}, \Sigma_{gen,j}\right), \tag{3.11}$$

$$\hat{f}_{imp}(\boldsymbol{s}) = \sum_{j=1}^{M_{imp}} p_{imp,j}\phi^K\left(\boldsymbol{s}; \boldsymbol{\mu}_{imp,j}, \Sigma_{imp,j}\right), \tag{3.12}$$

where $M_{gen}$ ($M_{imp}$) is the number of mixture components used to model the density of the genuine (impostor) scores, $\mu_{gen,j}$ ($\mu_{imp,j}$) and $\Sigma_{gen,j}$ ($\Sigma_{imp,j}$) are the mean vector and covariance matrix corresponding to the $j^{th}$ mixture component in $\hat{f}_{gen}(\boldsymbol{s})$ ($\hat{f}_{imp}(\boldsymbol{s})$) and $p_{gen,j}$ ($p_{imp,j}$) is the weight assigned to the $j^{th}$ mixture component in $\hat{f}_{gen}(\boldsymbol{s})$ ($\hat{f}_{imp}(\boldsymbol{s})$). In equations (3.11) and (3.12), the sum of the component weights is 1, i.e., $\sum_{j=1}^{M_{gen}} p_{gen,j} = 1$ and $\sum_{j=1}^{M_{imp}} p_{imp,j} = 1$.

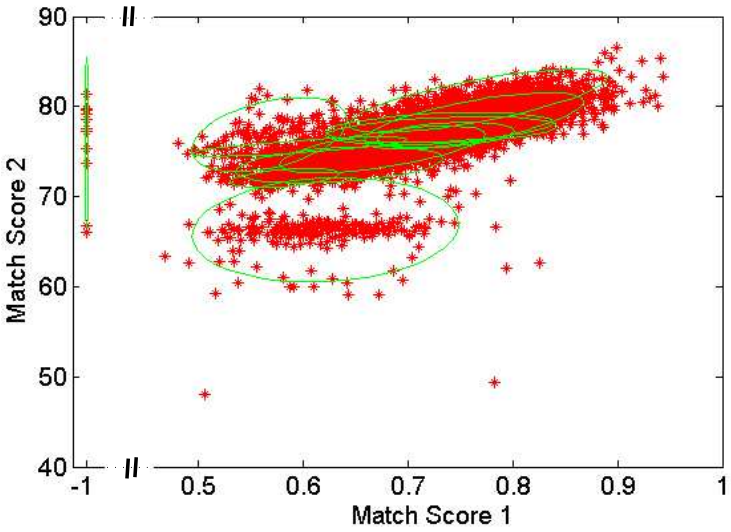
We use the algorithm proposed by Figueiredo and Jain [67] to estimate the parameters of the mixture densities in equations (3.11) and (3.12). Selecting the appropriate number of components is one of the most challenging issues in mixture density estimation; while a mixture with too many components may result in over-fitting, a mixture with too few components may not approximate the true density well. The GMM fitting algorithm proposed in [67] [4] automatically estimates the number of components and the component parameters using an EM algorithm and the minimum message length (MML) criterion. This algorithm is also robust to initialization of parameter values (mean vectors and covariance matrices) and can handle discrete components in the match score distribution by modeling the discrete scores as a mixture component

[4]The MATLAB code for this algorithm is available at `http://www.lx.it.pt/~mtf/mixturecode.zip`

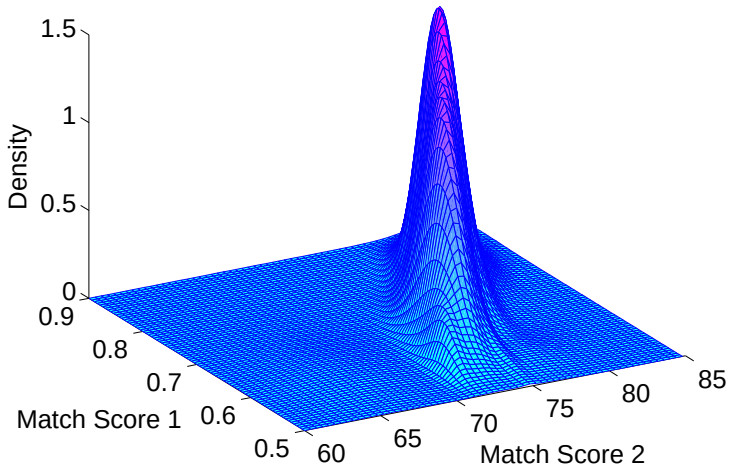
with very small variance. This is achieved by adding a small value (regularization factor) to the diagonal of the covariance matrices. The actual value of this variance does not affect the performance as long as it is insignificant compared to the variance of the continuous components in the match score distribution. For example, the lowest value of variance in the match score data used in our experiments is of the order of $10^{-3}$. Hence, we used the value of $10^{-5}$ as the lower bound for the variance. Our experiments indicate that a value smaller than $10^{-5}$ (say, $10^{-7}$ or $10^{-9}$) does not change the performance of GMM. Since we do not place any restrictions on the component covariance matrices $\Sigma_{gen,j}$ and $\Sigma_{imp,j}$, the estimates of the joint densities $\hat{f}_{gen}(\boldsymbol{x})$ and $\hat{f}_{imp}(\boldsymbol{x})$ also take into account the correlation between the match scores. Figures 3.6 and 3.7 show that Gaussian mixture model reliably estimates the 2-D genuine and impostor densities of the two face matchers in the NIST-Face database.

## 3.3 Incorporating Image Quality in Fusion

The quality of acquired biometric data directly affects the ability of a biometric matcher to perform the matching process effectively. Noise can be present in the biometric data due to defective or improperly maintained sensors, incorrect user interaction or adverse ambient conditions. For example, when noisy fingerprint images are processed by a minutiae based fingerprint recognition algorithm, a number of false (spurious) minutia points will be detected. Figures 3.8(c) and 3.8(d) show the minutiae extracted from good quality (Figure 3.8(a)) and noisy fingerprint (Figure

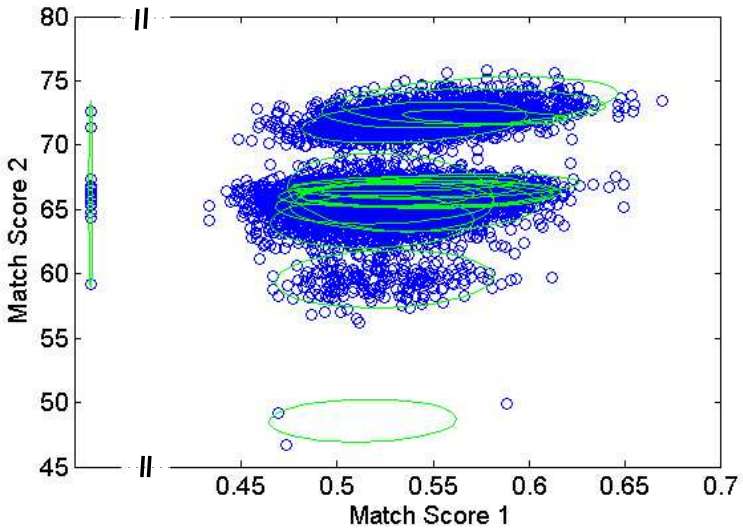



(a)

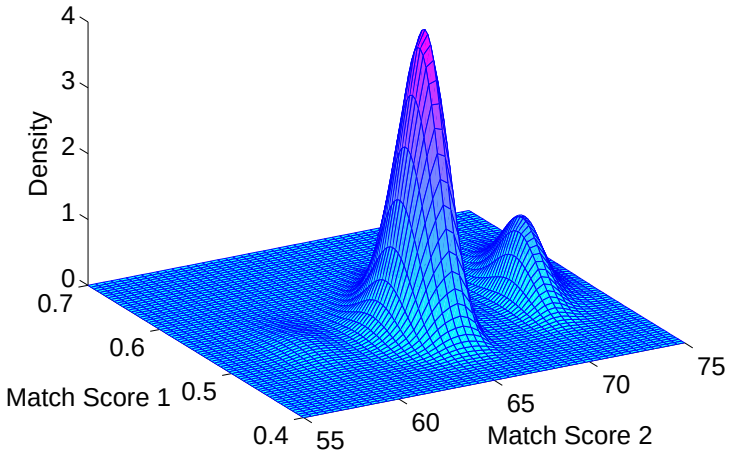



(b)

Figure 3.6: Density estimation based on Gaussian mixture models for the genuine scores in the NIST-Face database. (a) Scatter plot of the genuine scores along with the fitted mixture components and (b) density estimates of the genuine scores. In this case, 12 mixture components were found.




(a)




(b)

Figure 3.7: Density estimation based on Gaussian mixture models for the impostor scores in the NIST-Face database. (a) Scatter plot of the impostor scores along with the fitted mixture components and (b) density estimates of the impostor scores. In this example, 19 mixture components were found.

3.8(b)) images, respectively, using the minutiae extraction algorithm proposed in [86]. We observe that no false minutia is detected in the good quality fingerprint image shown in Figure 3.8(c). On the other hand, Figure 3.8(d) shows that several spurious minutiae are detected in the noisy image. In practice, some true minutiae may not be detected in poor quality images. These spurious and missing minutiae will eventually lead to errors in fingerprint matching [32].

Estimating the quality of a biometric sample and predicting the performance of a biometric matcher based on the estimated quality can be very useful in building robust multibiometric systems. This will allow us to dynamically assign weights to the individual biometric matchers based on the quality of the input sample to be verified. For example, consider a bimodal biometric system with iris and fingerprint as the two modalities. Let us assume that during a particular access attempt by the user, the iris image is of poor quality but the fingerprint image quality is sufficiently good. In this case, we can assign a higher weight to the fingerprint match score and a lower weight to the iris match score. With this motivation in mind, we now describe methods for automatically determining the quality of iris and fingerprint images and incorporating them into the fusion process.

To incorporate sample quality in the likelihood ratio framework, we first make the following observation. Since a poor quality sample will be difficult to classify as genuine or impostor (see Figure 3.9), the likelihood ratio for such a sample will be close to 1. On the other hand, for good quality samples, the likelihood ratio will be greater than 1 for genuine users and less than 1 for impostors. Hence, if we estimate the joint density of the match score and the associated quality, the resulting likelihood

(a) (b)

(c) (d)

Figure 3.8: Minutiae extraction results for fingerprint images of varying quality. (a) A good quality fingerprint image. (b) A noisy fingerprint image. (c) Minutia points detected in the good quality fingerprint image by an automatic minutiae extraction algorithm. (d) Minutia points detected in the noisy fingerprint image. The circles represent true minutia points while the squares represent false (spurious) minutiae. While no spurious minutia is detected in the good quality fingerprint image, several false minutia points are detected when the fingerprint image quality is poor.

ratios will be implicity weighted by the respective sample quality. We can still use the Gaussian mixture based density estimation technique described in section 3.2.2.

To perform quality-based fusion, we need to automatically extract quality information from the input biometric samples. Since biometric quality estimation is a challenging task in itself, we demonstrate the advantages of our scheme using only fingerprint and iris modalities for which quality estimators are readily available [32, 33]. However, the proposed quality-based fusion scheme is generic and can be applied to any biometric modality or matcher. Note that the match score depends on the quality of both the template and query samples, so we need to define a single quality index, known as *pairwise quality* [143], that takes into account the quality of both template and query images. We now describe techniques to compute the pairwise quality index for fingerprint and iris modalities.

### 3.3.1 Pairwise Fingerprint Quality

We estimate the local quality in a fingerprint image using the coherence measure described in [32]. Let $T_f$ and $Q_f$ represent the template and the query fingerprint images, respectively. We partition $T_f$ and $Q_f$ into blocks of size $12 \times 12$ pixels and estimate the coherence $\gamma$ and $\gamma'$ for each block in $T_f$ and $Q_f$, respectively. Let $\boldsymbol{M}_1, \ldots, \boldsymbol{M}_m$ be the $m$ minutiae in $T_f$, where $\boldsymbol{M}_i = \{x_i, y_i, \theta_i\}$, $i = 1, \ldots, m$. Let $\boldsymbol{M}'_1, \ldots, \boldsymbol{M}'_n$ be the $n$ minutiae in $Q_f$, where $\boldsymbol{M}'_j = \{x'_j, y'_j, \theta'_j\}$, $j = 1, \ldots, n$. Let $\gamma(x, y)$ and $\gamma'(x, y)$ be the quality (coherence) of the block which contains the location $(x, y)$ in $T_f$ and $Q_f$, respectively. Let $t(x, y, \Delta)$ be the rigid transformation function

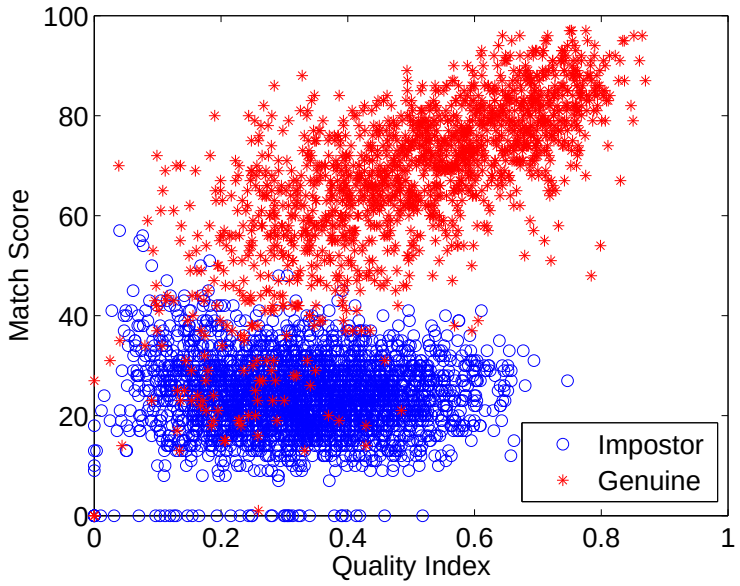



Figure 3.9: Variation of match score with quality for fingerprint modality in the WVU-Multimodal database. We observe that the genuine and impostor match scores are well-separated only for good quality (with quality index $> 0.5$) samples.

that transforms a point $(x, y)$ in $T_f$ to a point $(x', y')$ in $Q_f$. Here, $\Delta = [\Delta x, \Delta y, \Delta\theta]$ represents the translation and rotation parameters which are estimated using the 2-D dynamic programming based minutiae matcher described in [93]. Let $A$ and $A'$ be the area of the fingerprint regions in the template and the query. The area of overlap, $A_o$, between the fingerprint regions of $T_f$ and $Q_f$ can be computed using $\Delta$. The overall quality of the match between the template and query fingerprint images, $q_f(T_f, Q_f)$, is then defined as follows.

$$q_f(T_f, Q_f) = \left(\frac{r_1 + r_2}{m + n}\right)\left(\frac{2A_o}{A + A'}\right), \tag{3.13}$$

where

$$
\begin{aligned}
r_1 &= \sum_{i=1}^{m} \gamma(x_i, y_i)\gamma'(t(x_i, y_i, \Delta)) \text{ and} \\
r_2 &= \sum_{j=1}^{n} \gamma(t(x'_j, y'_j, -\Delta))\gamma'(x'_j, y'_j).
\end{aligned}
$$

Here, $0 \leq q_f(T_f, Q_f) \leq 1$. Note that if a minutia point in the template (query) falls outside the fingerprint region of the query (template) image, then the quality of that minutia is set to zero. Given good quality template and query fingerprint images with large overlap, $q_f(T_f, Q_f) \approx 1$.

### 3.3.2 Pairwise Iris Quality

We estimate the quality of match between the template and query iris images using a modified version of the wavelet-based iris quality assessment scheme proposed in [33]. The template ($T_i$) and query ($Q_i$) iris images are segmented into iris and non-iris regions [33]. A 2-D isotropic Mexican hat wavelet filter is applied to the iris regions of $T_i$ and $Q_i$ at three different scales $(0.5, 1.0, 2.0)$ and the product of the responses at the three scales is obtained. In order to account for the variations in the pupil dilation, iris size and rotation, the rubber sheet model proposed by Daugman [50] is used to normalize the wavelet responses. Let $w_{r,s}$ be the product of the wavelet responses at the $r^{th}$ radius ($r = 1, \ldots, R$) and $s^{th}$ angle ($s = 1, \ldots, S$) in $T_i$ and let $w'_{r,s}$ be the corresponding wavelet response in $Q_i$. The average wavelet response at each radius $r$ is computed as $w_r$ $(= \frac{1}{S}\sum_{s=1}^{S} w_{r,s})$ and $w'_r$ $(= \frac{1}{S}\sum_{s=1}^{S} w'_{r,s})$

in $T_i$ and $Q_i$, respectively. The quality of match between the template and query iris images, $q_i(T_i, Q_i)$, is defined as the correlation coefficient between the vectors $\boldsymbol{w} = [w_1, \ldots, w_R]$ and $\boldsymbol{w'} = [w'_1, \ldots, w'_R]$. Here, $-1 \leq q_i(T_i, Q_i) \leq 1$.

## 3.4 Likelihood Ratio Based Fusion Rules

Based on the likelihood ratio test described in section 3.1, we consider three fusion rules: (i) complete likelihood ratio based fusion, (ii) product fusion and (iii) quality-based product fusion. The complete likelihood ratio based fusion rule does not involve any assumptions about the match score densities. In this case, the joint density is directly estimated by fitting the Gaussian mixture model as outlined in section 3.2. Given a vector of match scores $\boldsymbol{s} = (s_1, \ldots, s_K)$ generated by $K$ matchers, the complete likelihood ratio fusion rule can be stated as,

Assign $\boldsymbol{s}$ to the genuine class if

$$CLR(\boldsymbol{s}) = \frac{\hat{f}_{gen}(\boldsymbol{s})}{\hat{f}_{imp}(\boldsymbol{s})} \geq \tau, \tag{3.14}$$

where $\tau$ is the decision threshold that is determined based on the specified FAR.

The product fusion rule can be used when the matchers are assumed to be independent. Here, the joint density of the match scores is estimated as the product of the marginal densities. For a vector of match scores $\boldsymbol{s} = (s_1, \ldots, s_K)$ generated by $K$ matchers, the product fusion rule is given by

Assign $\boldsymbol{s}$ to the genuine class if

$$PLR(\boldsymbol{s}) = \prod_{k=1}^{K} \frac{\hat{f}_{gen,k}(s_k)}{\hat{f}_{imp,k}(s_k)} \geq \tau, \tag{3.15}$$

where $\hat{f}_{gen,k}(\cdot)$ and $\hat{f}_{imp,k}(\cdot)$ are the marginal densities of the genuine and impostor scores of the $k^{th}$ matcher.

The quality-based product fusion rule assumes independence between the $K$ biometric matchers. However, within each biometric matcher the match score and the quality measure can be correlated. Let $q_k$ be the quality of the match provided by the $k^{th}$ matcher, for $k = 1, \ldots, K$. Let $\hat{f}_{gen,k}(s_k, q_k)$ ($(\hat{f}_{imp,k}(s_k, q_k))$ be the joint density of the match score and the quality estimated from the genuine (impostor) template-query pairs of the $k^{th}$ matcher. The quality-based product fusion rule is given by

Assign $\boldsymbol{s}$ to the genuine class if

$$QPLR(\boldsymbol{s}, \boldsymbol{q}) = \prod_{k=1}^{K} \frac{\hat{f}_{gen,k}(s_k, q_k)}{\hat{f}_{imp,k}(s_k, q_k)} \geq \tau. \tag{3.16}$$

It is also possible to compute the joint density of the $K$ match scores and $K$ quality values without assuming the independence of the matchers. However, we do not consider this rule because it requires estimating the joint density of a rather large number of variables ($K \times 2$), which may not be reliable with limited training data that is often encountered in practice.

The likelihood ratio based fusion framework can also be used for fusion of soft biometric information (e.g., gender, ethnicity and height) with the primary biometric identifiers (e.g., fingerprint and face). For instance, Jain et al. [89] used the product

fusion rule proposed here for fusion of soft and primary biometric traits. This requires computation of soft biometric likelihoods as described in [169].

## 3.5 Sequential Fusion Using Likelihood Ratio Framework

The likelihood ratio based fusion rules proposed in section 3.4 can be applied only in a multibiometric verification system operating in the parallel mode where the scores from all the matchers are available prior to fusion. However, in some applications a multibiometric system operating in the cascade or sequential mode (see section 2.3) may be more appropriate because a sequential system has higher throughput and is more convenient for the user. For example, when a cascaded multibiometric system has sufficient confidence on the identity of the user after processing the first biometric source, the user may not be required to provide the other sources of information.

One method to extend the likelihood ratio based fusion framework for a sequential multibiometric system is to employ the sequential probability ratio test (SPRT) [201]. At stage $k$ in a SPRT, the score ($s_k$) output by the $k^{th}$ matcher is used to compute the marginal likelihood ratio, $L_k$, where

$$L_k = \frac{\hat{f}_{gen,k}(s_k)}{\hat{f}_{imp,k}(s_k)}. \tag{3.17}$$

Here, $\hat{f}_{gen,k}(\cdot)$ and $\hat{f}_{imp,k}(\cdot)$ are the marginal densities of the genuine and impostor scores of the $k^{th}$ matcher and $k = 1, 2, \cdots, K$. The marginal likelihood ratio ($L_k$,

$k = 1, 2, \cdots K-1$) is compared to two different thresholds $A_k$ and $B_k$, where $A_k > B_k$. When $L_k > A_k$, we decide in favor of the genuine class. On the other hand, if $L_k < B_k$, we decide in favor of the impostor class. Only when $B_k \leq L_k \leq A_k$, the test proceeds to the next stage $(k+1)$. At stage $K$, if no decision has been made, the process can be truncated by setting $A_K = B_K$. While the SPRT is a principled approach to handle fusion in a cascade multibiometric system, it has the following limitations. Firstly, determining the optimal values of the thresholds $A_k$'s and $B_k$'s is not an easy task, particularly when the score densities do not have a simple parametric form. Secondly, the SPRT assumes that the sequence in which the matchers are to be invoked is fixed a priori. Finally, while Devijver and Kittler [53] have shown that it is possible to incorporate the cost of invoking a matcher when determining the thresholds in a SPRT, such an approach adds further complexity in the threshold determination process. Due to these reasons, we use a simple binary decision tree classifier [57] based on the marginal score densities of the individual matchers to extend the likelihood ratio framework for the sequential fusion scenario.

During the training phase, the marginal genuine and impostor score densities are estimated as described in section 3.2 and the marginal likelihood ratios of the training samples are obtained. The marginal likelihood ratios are treated as features and are used to train a binary decision tree classifier using the $C4.5$ decision tree learning algorithm [57]. During the authentication phase, the biometric modalities are acquired and the marginal likelihood ratios are computed in the order in which the different modalities appear in the decision tree starting from the root node. The main advantage of the decision tree based approach for sequential fusion is its simplicity in

terms of learning and implementation. However, the major limitation of this approach is that it is not straightforward to control the tree complexity (number of levels in the tree and positions of the leaf nodes). Since the goal of a cascade multibiometric system is to increase the throughput and user convenience, the number of levels in the tree should be small and the leaf nodes should be as close as possible to the top of the tree (especially for the genuine class), thereby favoring early decisions. Heuristic pruning approaches are needed to obtain a decision tree that satisfies the above two requirements.

## 3.6 Experimental Results

The performances of likelihood ratio based fusion rules were evaluated on two public-domain databases, namely, NIST-BSSR1 and XM2VTS-Benchmark databases. The performance of the quality-based product fusion rule was evaluated only on the WVU-Multimodal database since the other databases do not contain raw fingerprint and iris images to enable us to estimate the biometric sample quality. A description of these multibiometric databases can be found in the Appendix. Density estimates based on both the modified kernel density estimator and Gaussian mixture model-based estimator lead to almost identical fusion results on all the databases. Therefore, we report only the performance of GMM-based density estimation in the subsequent sections.

### 3.6.1 Evaluation Procedure

For each experiment, half of the genuine and impostor match scores were randomly selected to be in the training set for estimating the marginal densities and the correlation matrices[5]. The remaining genuine and impostor scores were used for analyzing the effectiveness of the fusion rules. The above training-test partitioning was repeated $m$ times ($m = 20$) and the reported ROC curves correspond to the mean GAR values over the $m$ trials at different FAR values.

The following procedure is used to test if the difference in performances of two different fusion algorithms is significant. Let $GA_i$ and $GB_i$ be the GAR of two different fusion rules $A$ and $B$, respectively, at a specific value of FAR for the $i^{th}$ trial, $i = 1, \cdots, m$. Let $D_i = (GA_i - GB_i)$ be the difference between the GAR values of the two rules for the $i^{th}$ trial and let $\mu_D$ be the expected difference. If we assume that $D_i$'s are independent and normally distributed with variance $\sigma_D^2$, then hypotheses about $\mu_D$ can be tested using a paired $t$ test [164]. To determine if the performance of rule $A$ is better than that of rule $B$, we test the null hypothesis $H_0$: $\mu_D \leq 0$ against the alternative hypothesis $H_1$: $\mu_D > 0$. Here, rejecting the null hypothesis indicates that the performance of rule $A$ is better than that of rule $B$. The test statistic is given by

$$t = \frac{\bar{D}}{s_D/\sqrt{m}}, \tag{3.18}$$

[5]For experiments on the XM2VTS-Benchmark database, we do not randomly partition the score data into training and test sets because this partitioning is already defined by the Lausanne Protocol-1 [154]. Hence, confidence intervals are not estimated for experiments with the XM2VTS-Benchmark database.

where $\bar{D}$ and $s_D$ are the sample mean and standard deviation, respectively, of the $D_i$'s, $i = 1, \cdots, m$. For an $\alpha$ level test, the null hypothesis must be rejected if $t \geq t_{(\alpha,m-1)}$, where $t_{(\alpha,m-1)}$ is the value such that a fraction $\alpha$ of the area under the $t$ distribution with $m-1$ degrees of freedom lies to the right of $t_{\alpha,m-1}$. The $100(1-\alpha)\%$ confidence interval for $\mu_D$ is given by $\bar{D} \pm t_{(\alpha/2,m-1)} s_D/\sqrt{m}$. Here, a $100(1-\alpha)\%$ confidence interval denotes that if the database is randomly partitioned into training and test sets a large number of times and if the confidence interval is estimated for these trials, then 95% of these confidence intervals would contain the true value of $\mu_D$. The value of $\alpha$ is set to 0.05 in our experiments.

### 3.6.2 Performance of Likelihood Ratio Based Parallel Fusion

The performance of complete likelihood ratio based fusion rule was evaluated on the three partitions of the NIST-BSSR1 database and the XM2VTS-Benchmark database. The receiver operating characteristic (ROC) curves of the individual matchers and the likelihood ratio based fusion rule for these databases are shown in Figures 3.10, 3.11, 3.12 and 3.13. As expected, likelihood ratio based fusion leads to significant improvement in the performance compared to the best single modality on all the four databases. At a false accept rate (FAR) of 0.01%, the improvement in the genuine accept rate (GAR) achieved due to likelihood ratio based fusion is presented in Table 3.1. We observe that the 95% confidence intervals estimated in Table 3.1 are fairly tight, which indicates that the performance improvement is consistent across different cross-validation trials.

Table 3.1: Performance improvement achieved due to likelihood ratio based fusion. The GAR values in the table correspond to 0.01% FAR.

| **Database** | **Best Single Matcher** | **Mean GAR** | | 95% **Confidence Interval on increase in GAR** |
|---|---|---|---|---|
| | | **Best Single Matcher** | **Likelihood Ratio based Fusion** | |
| NIST-Multimodal | Right Index Finger | 85.3% | 99.1% | [13.5%, 14%] |
| NIST-Fingerprint | Right Index Finger | 83.5% | 91.4% | [7.6%, 8.2%] |
| NIST-Face | Matcher 1 | 71.2% | 77.2% | [4.7%, 7.3%] |
| XM2VTS-Benchmark | DCTb-GMM Face Matcher | 89.5% | 98.7% | N/A |

### 3.6.3 Comparison With Other Score Fusion Techniques

The performance of the LR fusion rule is first compared to fusion based on Support Vector Machine (SVM) classifier. While the performance of SVM based fusion is comparable to LR fusion on the NIST-Fingerprint and XM2VTS-Benchmark databases (see Figures 3.11 and 3.13), it is inferior to LR fusion on the NIST-Multimodal and NIST-Face databases (see Figures 3.10 and 3.12). Moreover, the kernel function and the associated parameters for SVM must be carefully chosen in order to achieve this performance. For example, while linear SVM gave good performance on the NIST-Multimodal and XM2VTS-Benchmark databases, a radial basis function (RBF) kernel with different parameter values for the NIST-Fingerprint ($\gamma = 0.005$) and NIST-Face ($\gamma = 0.1$) databases was used to obtain the results reported here. In our experiments,

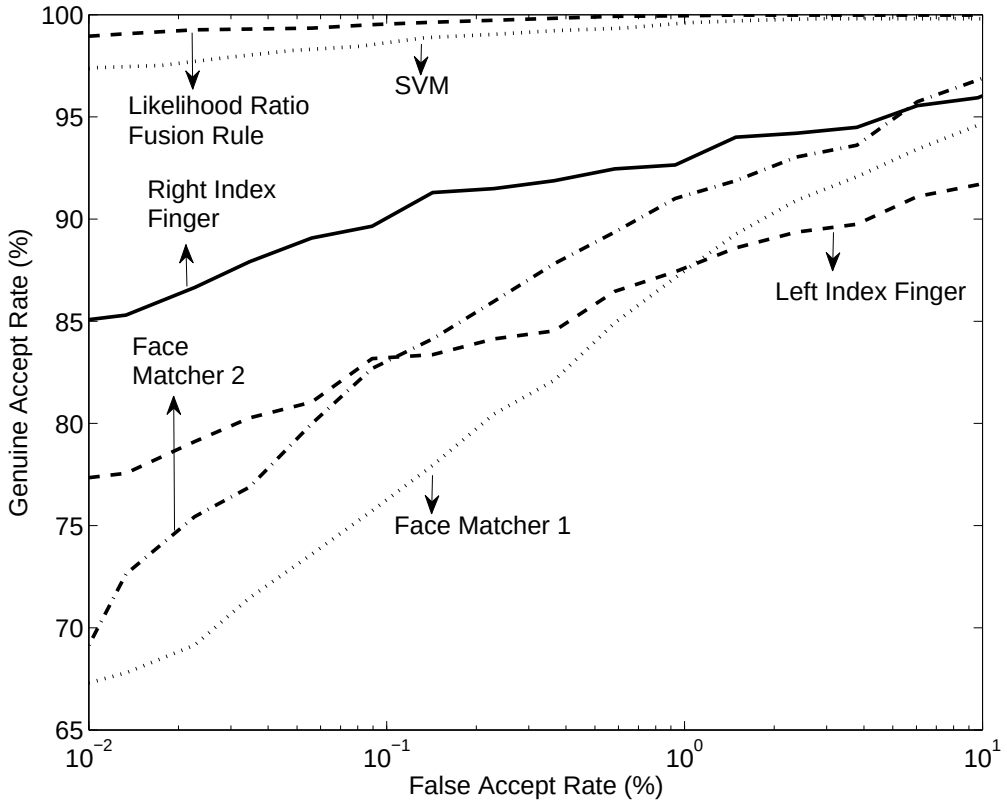



Figure 3.10: Performance of complete likelihood ratio based fusion rule and linear SVM-based fusion on the NIST-Multimodal database.

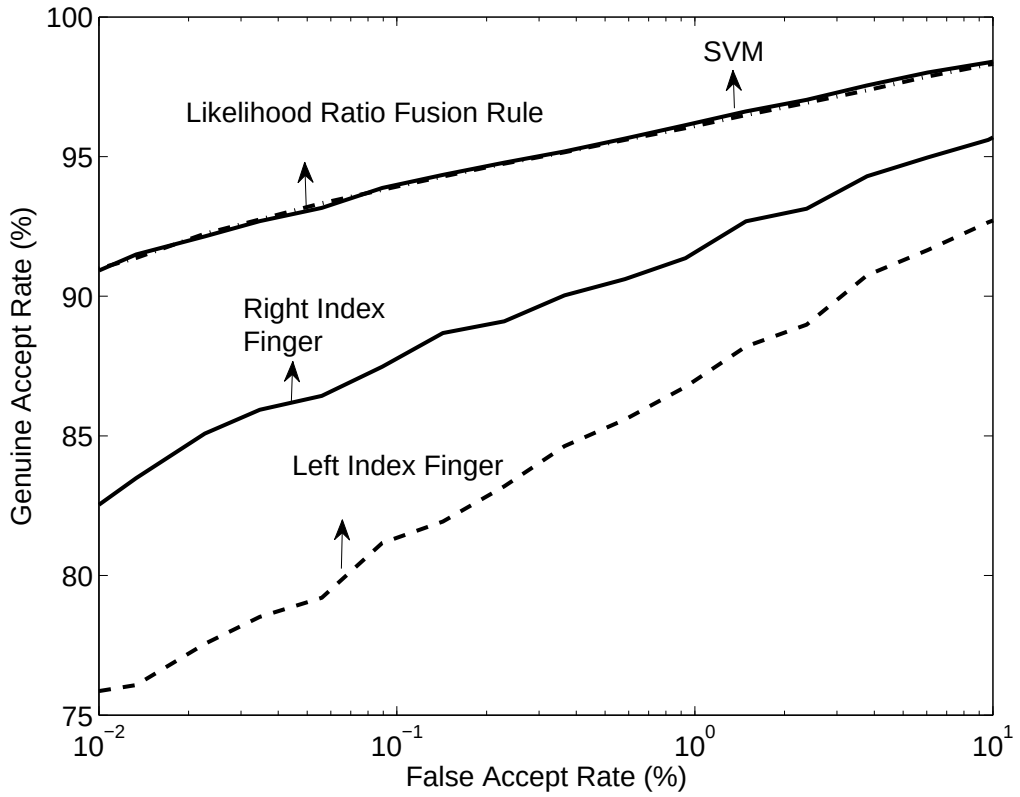



Figure 3.11: Performance of complete likelihood ratio based fusion rule and SVM-based fusion on the NIST-Fingerprint database. A radial basis function kernel with $\gamma = 0.005$ was used for SVM fusion.

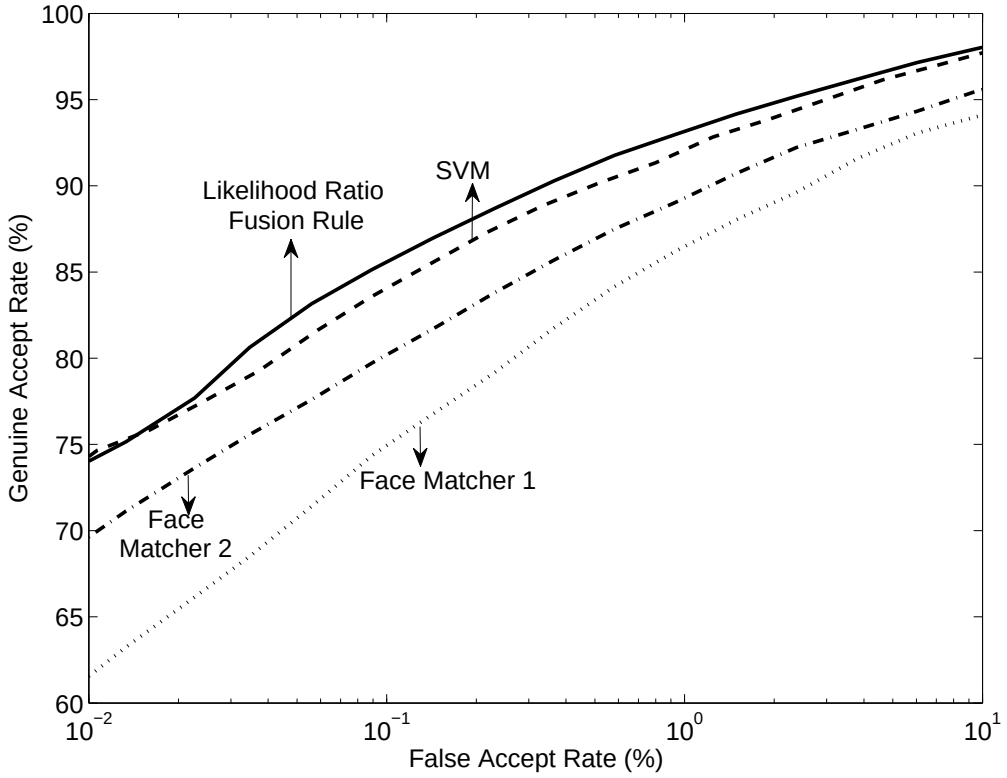



Figure 3.12: Performance of complete likelihood ratio based fusion rule and SVM-based fusion on the NIST-Face database. A radial basis function kernel with $\gamma = 0.1$ was used for SVM fusion.

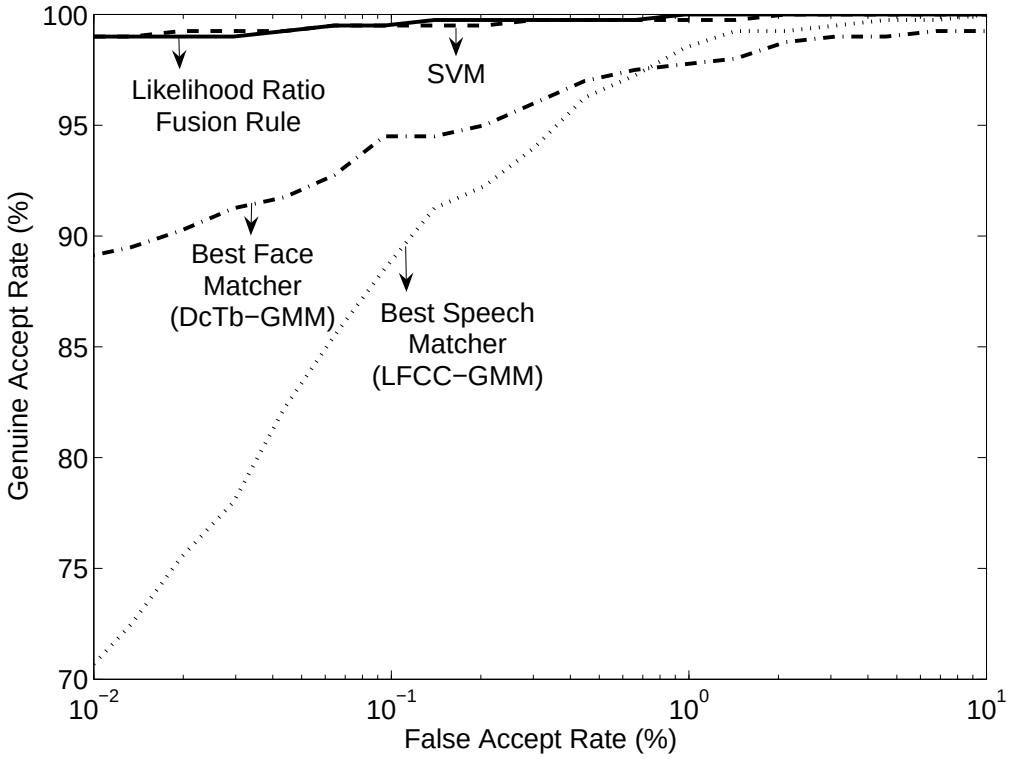
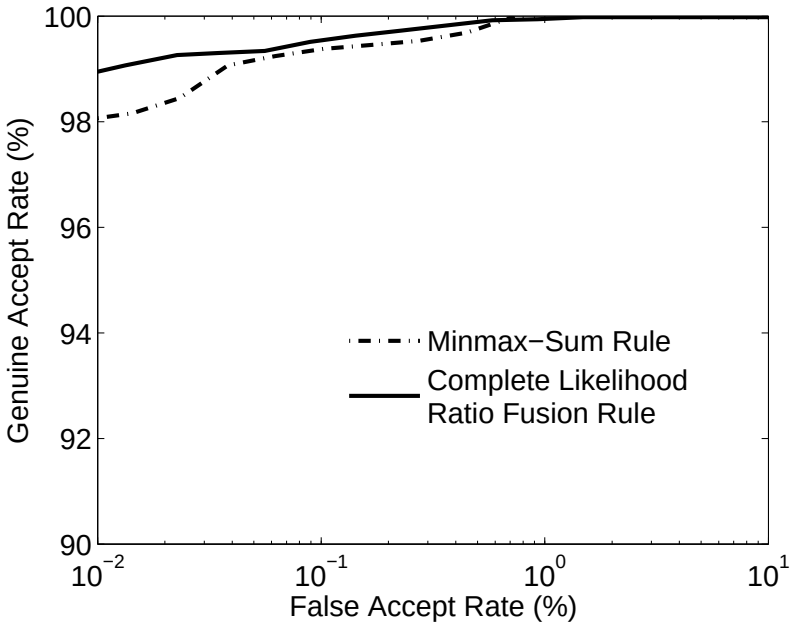



Figure 3.13: Performance of complete likelihood ratio based fusion rule and linear SVM-based fusion on the XM2VTS-Benchmark database. Although there are 8 different matchers in the XM2VTS-Benchmark database, only the ROC curves of the best face matcher (DCTb-GMM) and the best speech matcher (LFCC-GMM) are shown for clarity.

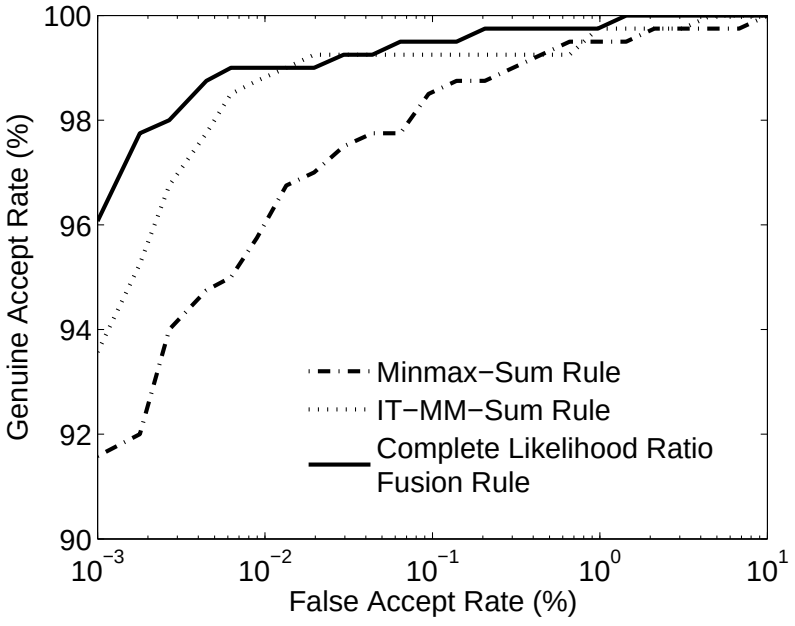
the model selection for SVM (kernel type and kernel parameters) was performed by trial and error. We manually tried the linear SVM and RBF kernel with different parameter choices (approximately 5 different values) on each database and report the best results. It is also possible to set the values of the kernel parameters automatically using techniques proposed in the literature [29, 70].

Next, we compare the performance of complete likelihood ratio based fusion rule with commonly used transformation-based score fusion techniques, where the scores are first transformed using a normalization scheme and then the normalized scores are combined using a fusion rule. Among the various possible combinations of normalization schemes and fusion rules [90, 180], we selected the min-max normalization scheme and sum of scores fusion method because our empirical results showed that this combination gave the best results. The ROC curves for the likelihood ratio based and sum of scores fusion rules on NIST-Multimodal and XM2VTS-Benchmark databases are shown in Figure 3.14. In the case of NIST-Multimodal database, we observe that the complete likelihood ratio based fusion rule does not provide any significant improvement over the sum rule (see Figure 3.14(a)). The paired $t$ test rejects the hypothesis that the performances of the likelihood and sum rules are different. This is not surprising, because it has been shown in the literature that the sum rule works quite well in practice due to its robustness to noisy data and errors in density estimation [108]. However, the performance of the sum rule is inferior to the likelihood ratio based approach in the case of XM2VTS-Benchmark database (see Figure 3.14(b)).

The reason for the sub-optimal performance of sum rule in the case of XM2VTS-




(a)



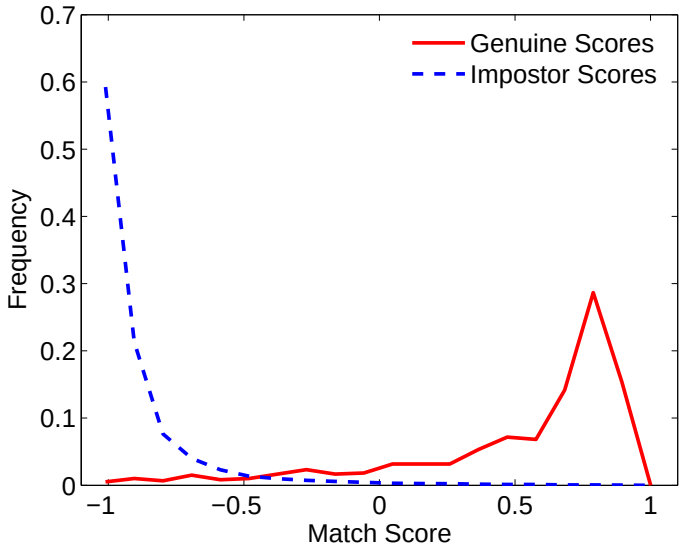

(b)

Figure 3.14: Performance of complete likelihood ratio based fusion rule and sum of scores fusion rule with min-max normalization on (a) NIST-Multimodal database and (b) XM2VTS-Benchmark database. In (b), IT-MM denotes that an inverse tangent function is applied only to the match scores of the MLP classifiers prior to normalizing all the match scores using min-max normalization.

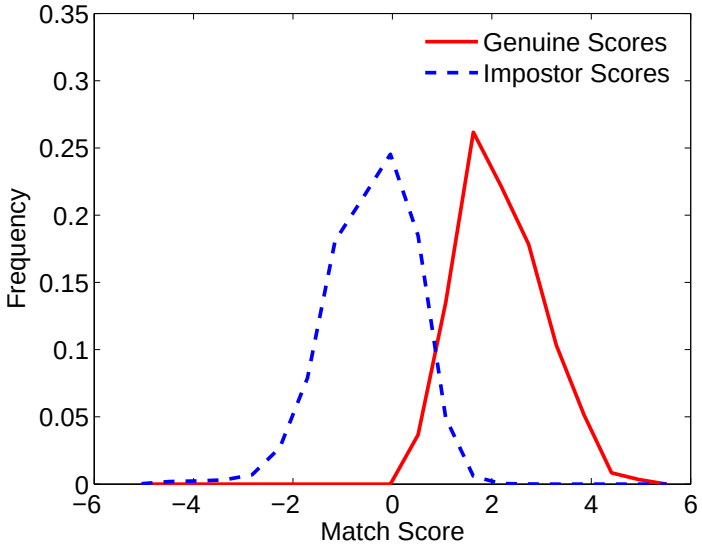
Benchmark database is that the match scores are computed based on two types of classifiers. One of them is a multi-layer perceptron (MLP) while the other is a Bayes classifier using the Gaussian Mixture Model (GMM). While the distribution of match scores output by the GMM classifier can be approximated by a Gaussian distribution (see Figure 3.15(b)), the match score distribution of the MLP classifier is peaked around 1 and $-1$ due to the tanh function at the output layer of the perceptron (see Figure 3.15(a)). Hence, the sum rule does not provide a good approximation to the likelihood ratio based fusion rule because the nature of match score distributions is very different. However, if we change the distribution of scores at the output of the MLP classifier by applying an inverse tangent function to these scores, then the performance of the sum rule improves and becomes comparable to likelihood ratio based fusion as observed in Figure 3.14(b). These results demonstrate that while it is possible to achieve good fusion performance for a specific database using the simple sum rule by carefully choosing the normalization scheme, the proposed likelihood-ratio based fusion framework is a general approach that provides good performance consistently on all the databases considered in this thesis.

### 3.6.4 Comparison of Product and Complete Likelihood Ratio Fusion

The complete likelihood ratio based fusion rule is based on the joint density of the genuine and impostor match scores and hence, takes into account the correlation between the matchers. On the other hand, the product fusion rule, which is simpler




(a)




(b)

Figure 3.15: Distribution of genuine and impostor match scores in the XM2VTS-Benchmark database for (a) MLP classifier and (b) GMM classifier.

to implement, ignores the correlation between matchers and approximates the joint density by the product of the marginal densities. To study the performance difference between these two rules, we consider two databases for which the correlation between the various matchers is high. As an example, in the NIST-Face database, the correlation between the scores of the two matchers is 0.7 for the genuine class and 0.3 for the impostor class. In the XM2VTS-Benchmark database, we choose the two speech matchers LFCC-GMM and SSC-GMM because this matcher pair had the highest correlation value among the different matcher pairs (0.8 for the genuine class and 0.7 for the impostor class).

The performance of the product and complete likelihood ratio based fusion rules on the NIST-Face database is shown in Figure 3.16, which indicates that there is no difference in the performance of the two rules. This is because the difference between genuine and impostor correlations is not high and the two matchers in this database are reasonably accurate (the $d^{'}$ value[6] is 3.2 for both the matchers). Now, we apply a linear transformation of the form $s_k^{'} = (s_k - a)/b$ to the genuine match scores from the two matchers, where $s_k$ is the original score of the $k^{th}$ matcher and $s_k^{'}$ is the modified score. The values of the constants $a$ and $b$ are chosen such that the $d^{'}$ metric of the transformed scores is approximately 2. This linear transformation does not affect the correlation between the genuine scores of the two matchers. We also remove the correlation between impostor scores by randomly permuting the impostor scores from one of the two matchers. Note that this permutation does not change the

[6]The d-prime value ($d^{'}$) measures the separation between the means of the genuine and impostor distributions in standard deviation units. A higher $d^{'}$ value indicates better performance.

marginal distribution of the impostor scores. As a result of these transformations, the $d'$ value for the modified match scores is approximately 2 and the correlation between the scores is 0.7 for the genuine class and 0 for the impostor class. The performance of the complete likelihood ratio based fusion and the product fusion rules on the modified scores is shown in Figure 3.16. Since the separation between the genuine and impostor distributions was reduced by applying a linear transformation to the genuine scores, the accuracy of the individual matchers and hence the fusion performance is reduced substantially. However, in this case we observe that the complete likelihood ratio based fusion rule clearly outperforms the product fusion rule. For example, at a FAR of 0.1%, the average improvement in the GAR is 2.7% and the 95% confidence interval for the difference in the GAR between the two rules is $[2.5\%, 2.9\%]$. This result indicates that modeling the correlation between the match scores, and hence the use of complete likelihood ratio fusion rule is justified only if the matchers are of low accuracy and the difference between genuine and impostor correlation is large.

Similar results were also obtained in the case of correlated matcher pairs in the XM2VTS-Benchmark database. Figure 3.17 shows the ROC curves for the fusion of LFCC-GMM and SSC-GMM speech matchers in the XM2VTS database. The $d'$ values for the LFCC-GMM and SSC-GMM matchers are approximately 4 and 3, respectively. From Figure 3.17, we observe that the complete likelihood ratio based fusion and product fusion rules perform equally well on this pair of matchers. However, if the $d'$ values of the two matchers are reduced by applying a linear transformation to the genuine scores and if the impostor correlation is removed, we observe that the complete likelihood ratio based fusion rule provides better fusion performance than

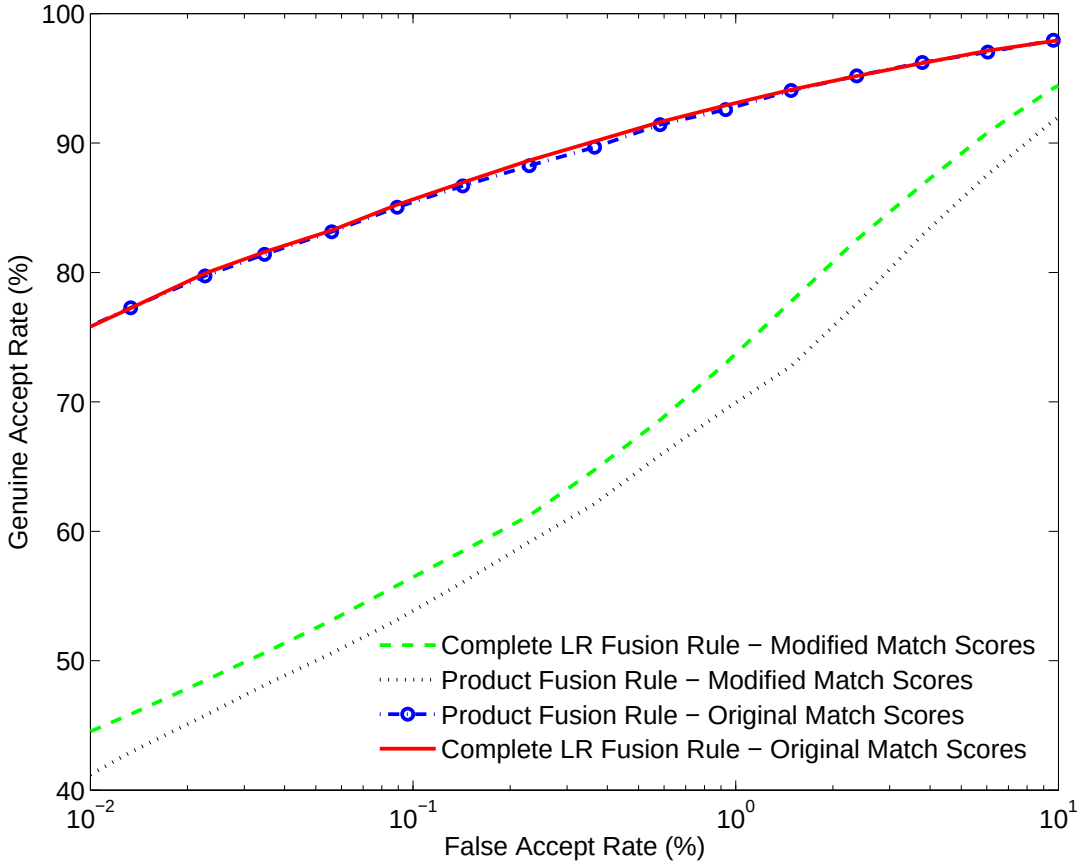



Figure 3.16: Performance of product and complete likelihood ratio based fusion rules for the two face matchers in the NIST-Face database.

the product fusion rule (see Figure 3.17).

### 3.6.5 Performance of Quality-based Fusion

We investigate the performance of the quality-based product fusion rule on the WVU-Multimodal database. Recall that for the other two databases, raw images are not available precluding the use of quality-based fusion. Figure 3.18 shows the performance of the product[7] and the quality-based product fusion rules. Fusion of fingerprint and iris modalities using the product rule gives a large improvement in the GAR compared to the best single modality (iris, in this experiment). The quality-based product fusion rule further improves the GAR. For example, at a FAR of 0.001%, the mean GAR of the iris modality is 66.7%, while the GAR values of the product and quality-based product fusion rules are 85.3% and 90%, respectively. The 95% confidence interval for the improvement in GAR obtained by using quality-based product fusion instead of product fusion is $[4.1\%, 5.3\%]$.

### 3.6.6 Performance of Likelihood Ratio Based Sequential Fusion

The performance of the decision tree based approach for likelihood ratio based sequential fusion was studied using the NIST-BSSR1 database. Since the structure of the decision tree depends on the set of match scores selected for training, the sequential fusion rule is not the same across all the cross-validation trials. A typical sequential

[7]Since the correlation between fingerprint and iris modalities is zero, complete likelihood ratio based fusion and product fusion rules have the same performance on the WVU-Multimodal database.

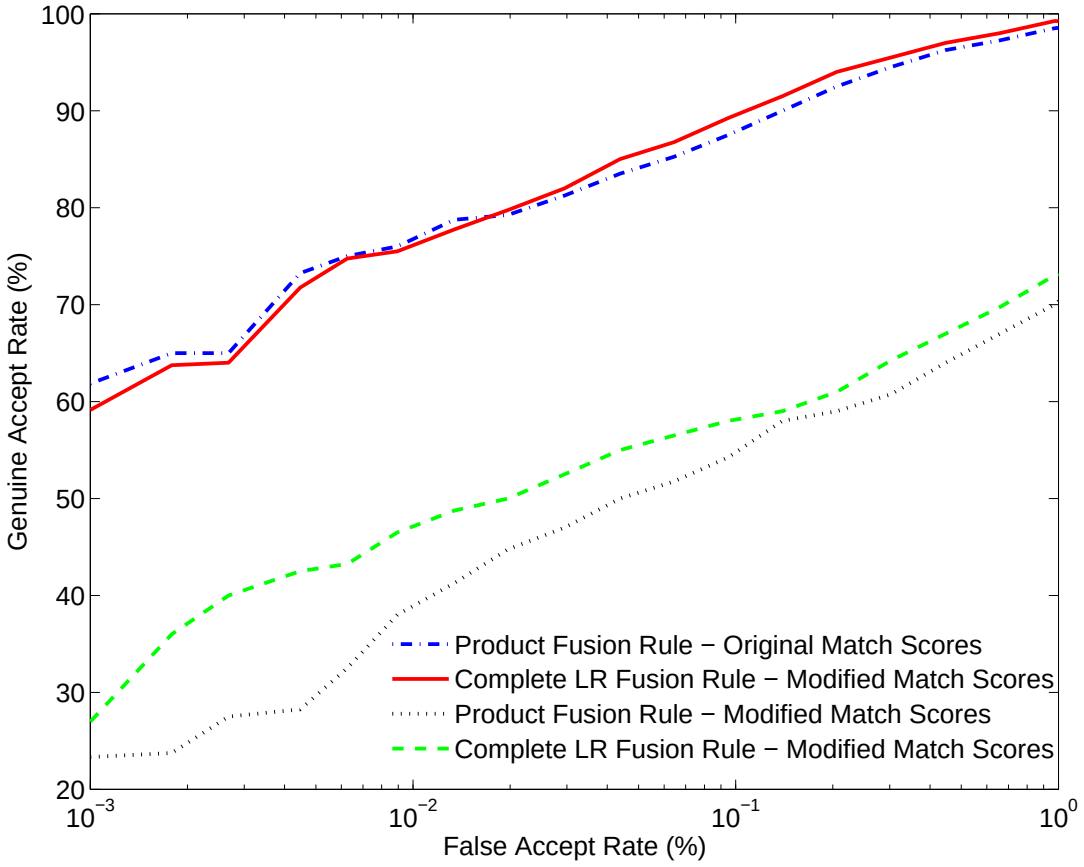



Figure 3.17: Performance of product and complete likelihood ratio based fusion rules for the LFCC-GMM and SSC-GMM speech matchers in the XM2VTS database.

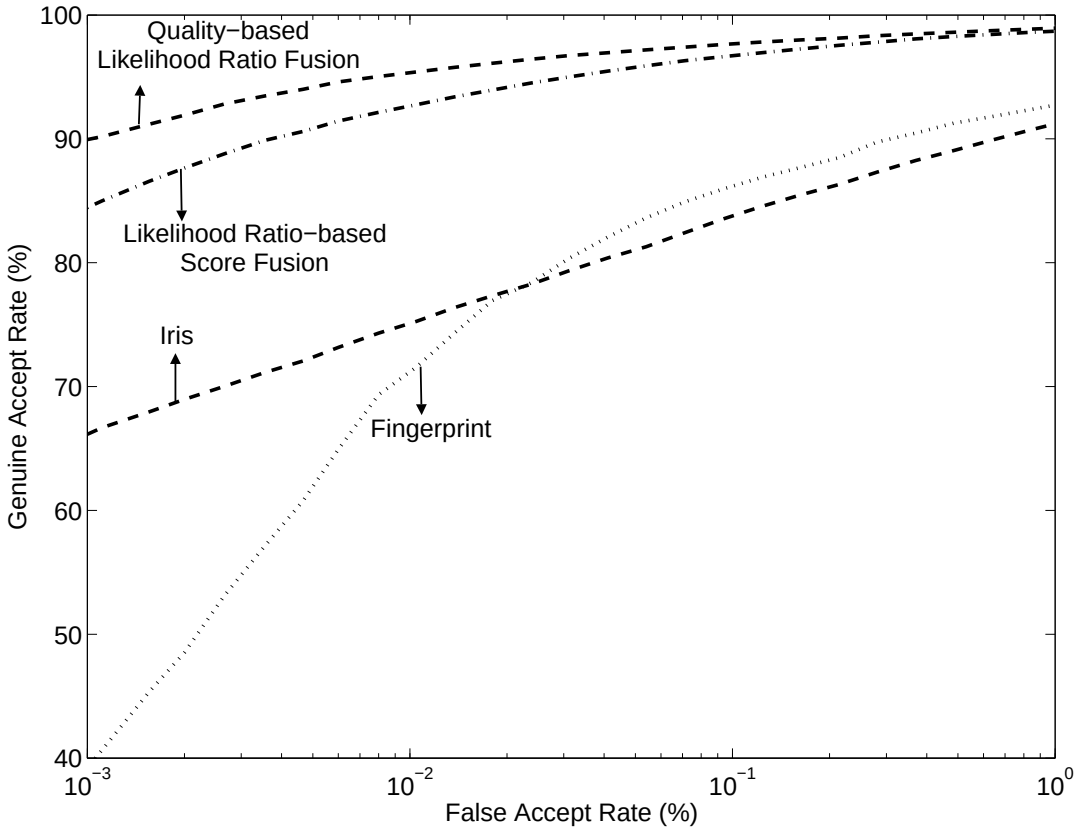



Figure 3.18: Performance of product fusion and quality-based product fusion rules on the WVU-Multimodal database.

fusion rule (decision tree) obtained using the NIST-Fingerprint database is shown in Figure 3.19. For this database, marginal likelihood ratio corresponding to the right index finger was usually selected as the root node because it is more accurate than the left index finger. On average, 92.6% of the genuine attempts required only a single modality (right index finger). The operating point of the system was modified by varying the ratio of genuine and impostor samples in the training phase. The average GAR of the system was observed to be 94.2% at a FAR of 0.2% and 95.9% at a FAR of 1.3%. The corresponding GAR values obtained in the parallel fusion scenario are 94.6% and 96.1%. These results show that while there is a marginal degradation in the GAR when sequential fusion is used instead of parallel fusion, the sequential system can lead to a significant increase in the user convenience and throughput because $\approx$ 92% of the genuine authentication attempts can be processed using just one modality.

Similar results were also observed in the case of the NIST-Multimodal database. Since both the face matchers in this database have roughly the same performance, we consider the scores from a single face matcher and the two fingers in this experiment. Figure 3.20 shows a typical sequential fusion rule (decision tree) obtained using the NIST-Multimodal database. Again in this database, the most accurate modality, namely, the right index finger was usually selected as the root node and on average, 91.1% of the genuine attempts required only a single modality (right index finger). The average GAR of the system was observed to be 96.9% at a FAR of 0.01% and 97.9% at a FAR of 0.2%. The corresponding GAR values obtained in the parallel fusion scenario are 97.8% and 98.6%. Thus, sequential fusion significantly reduces

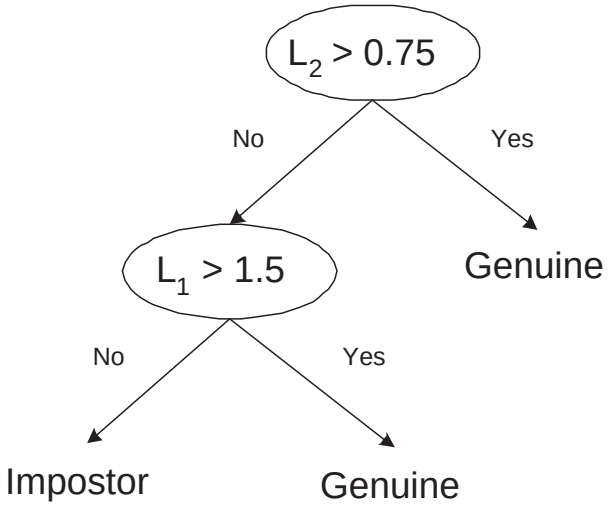



Figure 3.19: A typical sequential fusion rule (decision tree) obtained using the NIST-Fingerprint database. Here, $L_1$ and $L_2$ represent the marginal log-likelihood ratios for the left index finger and right index finger, respectively.

the number of modalities to be acquired during authentication without adversely affecting the GAR.

## 3.7 Summary

We have proposed a statistical framework for the fusion of match scores in a multibiometric verification system based on the likelihood ratio test. This approach is optimal provided the underlying genuine and impostor match score densities are known. In practice, one needs to estimate these densities from the available training set of match scores. We have modeled the genuine and impostor match scores using a mixture of Gaussian densities and used the EM algorithm with the minimum message length criterion for estimating the parameters of the mixture density and the number of mixture components. We have also developed a quality-based fusion scheme within the likeli-

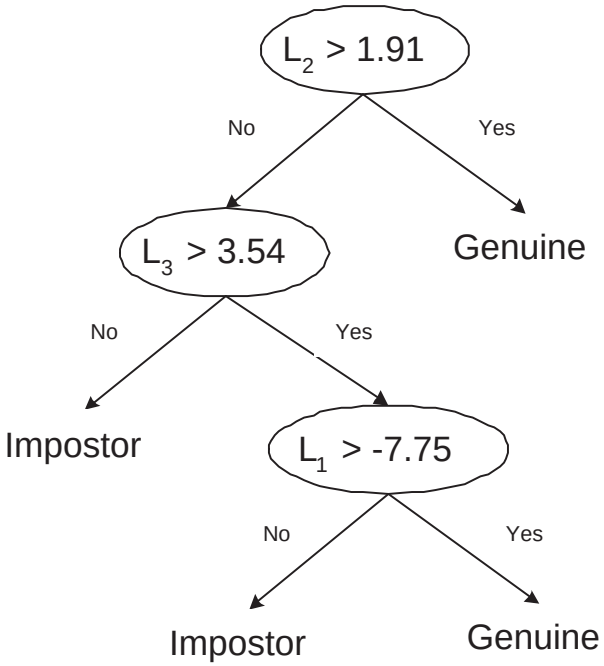



Figure 3.20: A typical sequential fusion rule obtained using the NIST-Multimodal database. Here, $L_1$, $L_2$ and $L_3$ represent the marginal log-likelihood ratios for the left index finger, right index finger and face modalities, respectively.

hood ratio framework to fuse multiple biometric sources based on the input biometric sample quality. Finally, we have shown that sequential fusion rules for a cascade multibiometric system can be generated by constructing a binary decision tree classifier based on the marginal likelihood ratio of the individual matchers. Experiments on three different multibiometric databases lead us to the following conclusions.

- Both the modified kernel density estimator and Gaussian mixture models provide reliable density estimates. However, The GMM-based density estimation is simpler to implement than KDE. The likelihood ratio based fusion rule based on the density estimates provided by GMM achieves consistently high recognition rates without any tuning of parameters by the system designer.

- The performance of a simple fusion rule such as the sum rule with min-max normalization is often comparable to that of the likelihood ratio based fusion rule. However, the sum rule requires careful selection of normalization scheme and fusion weights to achieve good performance. Further, this selection of normalization scheme and fusion weights is data dependent.

- In practice, the assumption of independence between matchers to be used does not adversely affect the performance of the fusion scheme, especially when the individuals matchers are quite accurate (equal error rate is less than 5%). In other words, the complete likelihood ratio fusion rule and the product likelihood ratio fusion rule give comparable performance.

- Utilizing biometric sample quality information, when available, in the likelihood ratio based fusion framework leads to a significant improvement in the

performance of multibiometric systems.

- The sequential fusion rules significantly reduce the number of modalities required for authentication and hence, increase the throughput and user convenience without degrading the recognition performance significantly.

# Chapter 4

# Multibiometric Identification

The likelihood ratio based score fusion framework proposed in Chapter 3 was developed specifically for the verification scenario where the goal is to decide whether an input sample belongs to the genuine or impostor class. In verification, the biometric query is compared only to the template of the claimed identity, resulting in a single match score for each matcher. However, in an identification system, the biometric query is compared with all the templates in the database resulting in $N$ match scores for each matcher, where $N$ is the number of persons enrolled in the database. The goal is to determine the true identity $I$ of the user based on these $N$ match scores, where $I \in \{I_1, I_2, \cdots, I_N, I_{N+1}\}$. Here, $I_1, I_2, \cdots, I_N$ correspond to the identities of the $N$ persons enrolled in the system and $I_{N+1}$ indicates the "reject" option, which is output when no suitable identity can be determined for the given query. When the reject option is available to the system, the problem is known as *open set* identification. On the other hand, if the biometric system is forced to make a decision in favor of one of the $N$ identities, then the problem is referred to as *closed set* identification.

In this chapter, we show that likelihood ratio based score fusion framework developed for the verification scenario is also applicable to multibiometric identification under certain assumptions. We also demonstrate that likelihood ratio based score fusion achieves good identification performance compared to other score level and rank level fusion approaches.

## 4.1 Score Level Fusion

Let $K$ denote the number of matchers in the multibiometric system and $N$ be the number of persons enrolled in the system. Let $S_n^k$ denote the random variable corresponding to the match score output by the $k^{th}$ matcher after comparing the query to the template of the $n^{th}$ person in the database, $k = 1, 2, \cdots, K;\ n = 1, 2, \cdots, N$. Let $\boldsymbol{\mathcal{S}}$ be a $N \times K$ matrix defined as

$$\boldsymbol{\mathcal{S}} = \begin{bmatrix} S_1^1 & \dots & S_1^k & \dots & S_1^K \\ \dots & & & & \\ S_n^1 & \dots & S_n^k & \dots & S_n^K \\ \dots & & & & \\ S_N^1 & \dots & S_N^k & \dots & s_N^K \end{bmatrix} = \left[\boldsymbol{S}^1, \boldsymbol{S}^2, \cdots, \boldsymbol{S}^K\right] = [\boldsymbol{S}_1, \boldsymbol{S}_2, \cdots, \boldsymbol{S}_N]^T,$$

where $\boldsymbol{S}^k = \left[S_1^k, S_2^k, \cdots, S_N^k\right]^T$, for $k = 1, 2, \cdots, K$ and $\boldsymbol{S}_n = \left[S_n^1, S_n^2, \cdots, S_n^K\right]$, for $n = 1, 2, \cdots, N$.

Suppose for a given query, we observe the $N \times K$ score matrix $\boldsymbol{s} = \left[s_n^k\right]$. Note

that $s_n^k$ represents the match score output by the $k^{th}$ matcher for the $n^{th}$ template in the database, $k = 1, 2, \cdots, K;\ n = 1, 2, \cdots, N$. Our goal is to determine the true identity $I$ of the given query based on $\boldsymbol{s}$. According to the Bayesian decision theory [57], the query should be assigned to the identity $I_{n_0}$ that maximizes the posteriori probability, i.e.,

$$\text{Assign} \quad I \rightarrow I_{n_0} \quad \text{if}$$

$$P(I_{n_0}|\boldsymbol{s}) \geq P(I_n|\boldsymbol{s}), \forall\ n = 1, 2, \cdots, N. \tag{4.1}$$

The above decision rule applies only to closed set identification. For open set identification, the query is assigned to identity $I_{n_0}$ only when equation (4.1) holds and $P(I_{n_0}|\boldsymbol{s}) \geq \tau$, where $\tau$ is a threshold.

We can estimate the posteriori probabilities $P(I_n|\boldsymbol{s})$ in the following manner. According to the Bayes theorem,

$$P(I_n|\boldsymbol{s}) = \frac{p(\boldsymbol{s}|I_n)P(I_n)}{p(\boldsymbol{s})}, \tag{4.2}$$

where $p(\boldsymbol{s}|I_n)$ is the likelihood of observing the score matrix $\boldsymbol{s}$ given that the true identity is $I_n$ and $P(I_n)$ is the prior probability of observing the identity $I_n$. If we assume equal prior for all the identities (i.e., $P(I_n) = 1/N, \forall\ n = 1, 2, \cdots, N$), the posteriori probability $P(I_n|\boldsymbol{s})$ is proportional to the likelihood $p(\boldsymbol{s}|I_n)$. Hence, we can rewrite the decision rule in equation (4.1) as

$$\text{Assign} \quad I \rightarrow I_{n_0} \quad \text{if}$$

$$p(\boldsymbol{s}|I_{n_0}) \geq p(\boldsymbol{s}|I_n), \forall\ n = 1, 2, \cdots, N. \tag{4.3}$$

Ideally, we would like to estimate the conditional density of $\boldsymbol{s}$ individually for each user because it captures the complete information about dependencies between the scores assigned to the different users and the user-specific characteristics of the match scores. However, directly estimating the conditional density of $\boldsymbol{s}$ is not practical due to the following two reasons.

1. Since $\boldsymbol{s}$ is a $N \times K$ dimensional matrix and $N$ is usually quite large (can be of the order of millions), estimating the density of $\boldsymbol{s}$ requires a significant number of training samples for each user, which is not generally available in multibiometric databases. Often, only a single template and query is available for each user.

2. The density of $\boldsymbol{s}$ needs to be re-estimated whenever there is a change in the list of persons enrolled in the biometric system, which may occur frequently.

Two simplifying assumptions are generally used [12, 75] to make the density estimation feasible. Firstly, we assume that the match scores for different persons are independent of one another. In other words, $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$ are assumed to be independent for all $i \neq j, i = 1, 2, \cdots; N, j = 1, 2, \cdots, N$. Based on this assumption, the likelihood $p(\boldsymbol{s}|I_n)$ can be simplified as

$$p(\boldsymbol{s}|I_n) = \prod_{j=1}^{N} p(\boldsymbol{s}_j|I_n) = p(\boldsymbol{s}_n|I_n) \prod_{j=1, j\neq n}^{N} p(\boldsymbol{s}_j|I_n). \tag{4.4}$$

Here, $p(\boldsymbol{s}_n|I_n)$ represents the density of genuine match scores corresponding to user $I_n$ and $p(\boldsymbol{s}_j|I_n), j \neq n$ represents the densities of the impostor scores.

The second assumption made is that the genuine match scores of all users are identically distributed, i.e., $p(\boldsymbol{s}_n|I_n) = p(\boldsymbol{s}_n|genuine) = f_{gen}(\boldsymbol{s}_n), \forall\ n = 1, 2, \cdots, N$ and the impostor match scores of all users are identically distributed, i.e., $p(\boldsymbol{s}_j|I_n) = p(\boldsymbol{s}_j|impostor) = f_{imp}(\boldsymbol{s}_j), \forall\ j, n = 1, 2, \cdots, N, n \neq j$. Therefore, equation (4.4) can be further rewritten as

$$p(\boldsymbol{s}|I_n) = f_{gen}(\boldsymbol{s}_n) \prod_{j=1, j\neq n}^{N} f_{imp}(\boldsymbol{s}_j). \tag{4.5}$$

Multiplying and dividing equation (4.5) by $f_{imp}(\boldsymbol{s}_n)$, we get

$$p(\boldsymbol{s}|I_n) = \frac{f_{gen}(\boldsymbol{s}_n)}{f_{imp}(\boldsymbol{s}_n)} \prod_{j=1}^{N} f_{imp}(\boldsymbol{s}_j). \tag{4.6}$$

Under the above two simplifying assumptions, the likelihood of observing the score matrix $\boldsymbol{s}$ given that the true identity is $I_n$ is proportional to the likelihood ratio that was used in the verification scenario. Thus, the decision rule in equation (4.3) can be restated as

$$\text{Assign} \quad I \rightarrow I_{n_0} \quad \text{if}$$

$$\frac{f_{gen}(\boldsymbol{s}n_0)}{f_{imp}(\boldsymbol{s}n_0)} \geq \frac{f_{gen}(\boldsymbol{s}n)}{f_{imp}(\boldsymbol{s}n)}, \forall\ n = 1, 2, \cdots, N. \tag{4.7}$$

## 4.2 Rank Level Fusion

When a biometric system operates in the identification mode, for a given query, the output of the system can be viewed as a ranking of the enrolled identities. In other words, the output indicates the set of possible matching identities sorted in a decreasing order of match scores. Although the ranks are derived from the match scores, the rank information captures the relative ordering of the scores corresponding to different users. The goal of rank level fusion schemes is to consolidate the ranks output by the individual biometric subsystems in order to derive a consensus rank for each identity.

Let $K$ denote the number of matchers in the multibiometric system and $N$ be the number of persons enrolled in the system. Suppose for a given query, we observe the $N \times K$ rank matrix $\boldsymbol{r} = \left[r_n^k\right]$, where $r_n^k$ represents the rank output by the $k^{th}$ matcher for the $n^{th}$ template in the database, $k = 1, 2, \cdots, K;\ n = 1, 2, \cdots, N$. The goal in rank level fusion is to determine the true identity $I$ of the given query based on $\boldsymbol{r}$. Let $r_n^{'}$ be a statistic computed for user $n$ such that the user with the lowest value of $r^{'}$ is assigned the highest consensus (or reordered) rank. For example, in the highest rank method [79], each user is assigned the highest rank (minimum $r$ value) as computed by different matchers, i.e., the statistic for user $n$ is

$$r_n^{'} = \min_{k=1}^{K} r_n^k. \tag{4.8}$$

Ties are broken randomly to arrive at a strict ranking order based on the new statistic $r^{'}$. Ho et al. [79] proposed other methods such as Borda Count and logistic regression which compute the statistic $r^{'}$ as a linear combination of ranks provided by the individual matchers. Melnik et al. [135] proposed the use of non-linear functions to combine the ranks of the individual matchers.

We now propose a new rank combination statistic based on Bayesian decision theory. Let $P_k(r)$ be the probability that the identity that is assigned rank $r$ by the $k^{th}$ matcher is the true identity, $r = 1, 2, \cdots, N;\ k = 1, 2, \cdots, N$. Note that the cumulative distribution function of the discrete rank distribution $P_k(r)$ is nothing but the Cumulative Match Characteristic (CMC) defined in section 1.3. Grother and Phillips [75] and Bolle et al. [12] show that the rank distribution $P_k(r)$ can be estimated provided the marginal genuine and impostor match score densities $f_{gen,k}(\cdot)$ and $f_{imp,k}(\cdot)$ are known. This estimation again requires the same two assumptions used in section 4.1, namely, (i) scores of the individual users are independent and (ii) genuine score distributions of different users are identical and the impostor score distributions of different users are identical.

For a given query, suppose that the identity $I_n$ is assigned the rank $r_n^k$ by the $k^{th}$ matcher. From the definition of the rank distribution $P_k(r)$, $P_k(r_n^k)$ is the posteriori probability that $I_n$ is the true identity given $r_n^k$. Further, if we assume that the matchers are independent, we can compute the new rank combination statistic as the product of the posterior probabilities of the individual matchers.

$$r_n^{'} = \prod_{k=1}^{K} P_k(r_n^k), \text{for} n = 1, 2, \cdots, N. \tag{4.9}$$

Note that for the rank statistic computed using equation (4.9), the user with the largest value of $r^{'}$ should be assigned the highest consensus rank. The rank posterior based fusion rule can then be defined as follows.

$$\text{Assign} \quad I \rightarrow I_{n_0} \quad \text{if}$$

$$r_{n_0}^{'} \geq r_n^{'}, \forall\ n = 1, 2, \cdots, N. \tag{4.10}$$

Note that likelihood ratio based score fusion rule shown in equation (4.7) utilizes only the match scores corresponding to a particular user, when computing the likelihood ratio for that user. In other words, the relative information between the scores of different users is ignored when computing the score likelihood ratio. On the other hand, the rank posterior based fusion rule in equation (4.10) considers only the relative order information between the scores of different users and the actual score values are ignored. Therefore, we can treat the score and rank information as two different pieces of evidence and define a hybrid fusion scheme that utilizes both the match scores and the ranks. Let $R$ the combined score and rank statistic, defined as

$$R_n(\boldsymbol{s}, \boldsymbol{r}) = P(I_n|\boldsymbol{s}) r_n^{'}, \tag{4.11}$$

where the posterior probability based on the match score matrix $\boldsymbol{s}$, $P(I_n|\boldsymbol{s})$, is computed by substituting equation (4.6) in equation (4.2) and the posterior probability

based on the rank matrix $\boldsymbol{r}$ is obtained using equation (4.9). The hybrid score-rank fusion rule can then be defined as

$$\text{Assign} \quad I \rightarrow I_{n_0} \quad \text{if}$$

$$R_{n_0} \geq R_n, \forall\ n = 1, 2, \cdots, N. \tag{4.12}$$

## 4.3 Experimental Results

The identification performance of various score and rank level fusion strategies was evaluated on the three partitions of the NIST-BSSR1 database. The cumulative match characteristic (CMC) curves of the individual matchers and the highest rank and hybrid score-rank fusion rules on the NIST-BSSR1 database are shown in Figures 4.1, 4.2 and 4.3. Similar to the verification scenario, in each experiment, half the users were randomly selected to be in the training set for estimating the marginal densities and the rank distribution. The remaining half of the database was used for evaluating the fusion performance. The above training-test partitioning was repeated 20 times and the reported CMC curves correspond to the mean identification rates over the 20 trials.

Among the various rank level fusion schemes such as highest rank, Borda count and logistic regression, we observed that the highest rank method achieves the best rank-m recognition rate when $m \geq K$, where $K$ is the number of matchers. Hence, only the recognition rates of the highest rank method are reported here. It is well-

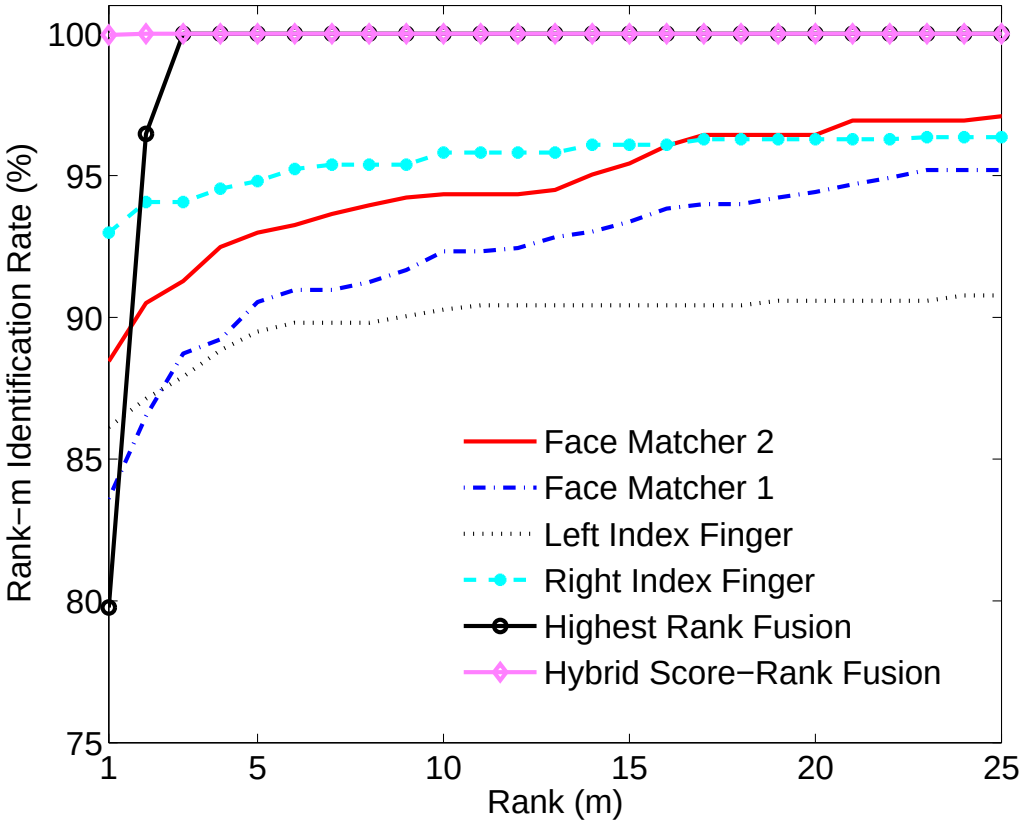



Figure 4.1: Cumulative Match Characteristic (CMC) curve of highest rank fusion and the hybrid score-rank fusion rules on the NIST-Multimodal database ($K = 4, N = 517$).

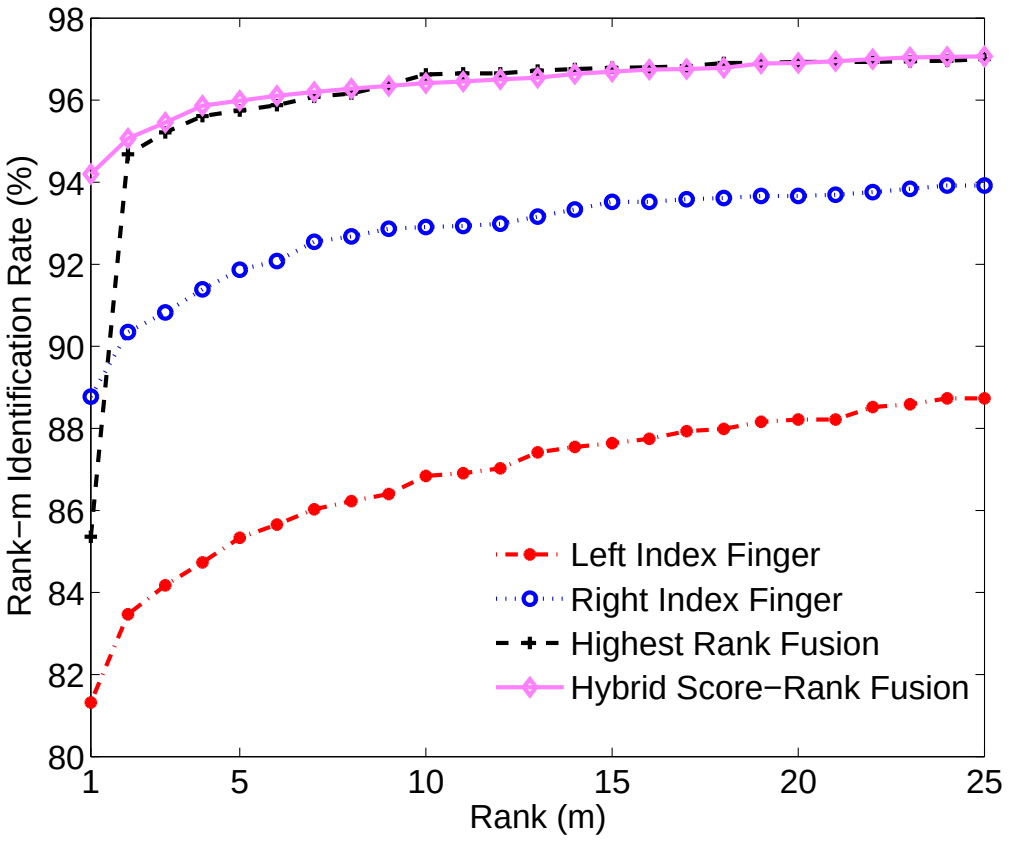



Figure 4.2: Cumulative Match Characteristic (CMC) curve of highest rank fusion and the hybrid score-rank fusion rules on the NIST-Fingerprint database ($K = 2, N = 6,000$).

known that the highest rank method works well when the number of users is large compared to the number of matchers [79], which is usually the case in biometric identification systems. This is because the highest rank method utilizes the strength of each matcher effectively. Even if only one matcher assigns a high rank to the correct user, it is still very likely that the correct user will receive a high rank after reordering. However, there can be up to $K$ ties at rank 1 due to conflicting decisions output by the $K$ matchers. Since the ties are broken randomly without considering the relative accuracies of the matchers, the identification rate of the highest rank method at ranks 1 to $K-1$ is not very high. In fact, the rank-1 accuracy of the highest rank method is usually less than the rank-1 accuracy of the best individual matcher.

The recognition rates of the likelihood ratio based score fusion rule, the rank posterior fusion rule and the hybrid score-rank fusion rule were observed to be quite similar on all the three partitions of NIST-BSSR1. While the hybrid score-rank fusion rule achieves a marginal improvement in the recognition rates over the other two fusion rules, the differences in the recognition rates of the three fusion rules is less than 1% at all ranks. Therefore, only the performance of the hybrid score-rank fusion rule is reported in Figures 4.1, 4.2 and 4.3. In the case of the NIST-Multimodal database, the hybrid score-rank fusion rule provides 100% rank-1 accuracy, while the rank-1 accuracy of the best single matcher (right index finger) was only 93.7%. The hybrid score-rank fusion rule improves the rank-1 accuracy from 88.9% for the best single matcher (right index finger) to 94% on the NIST-Fingerprint database. Finally, on the NIST-Face database the improvement is comparatively lower (81.2% for the best

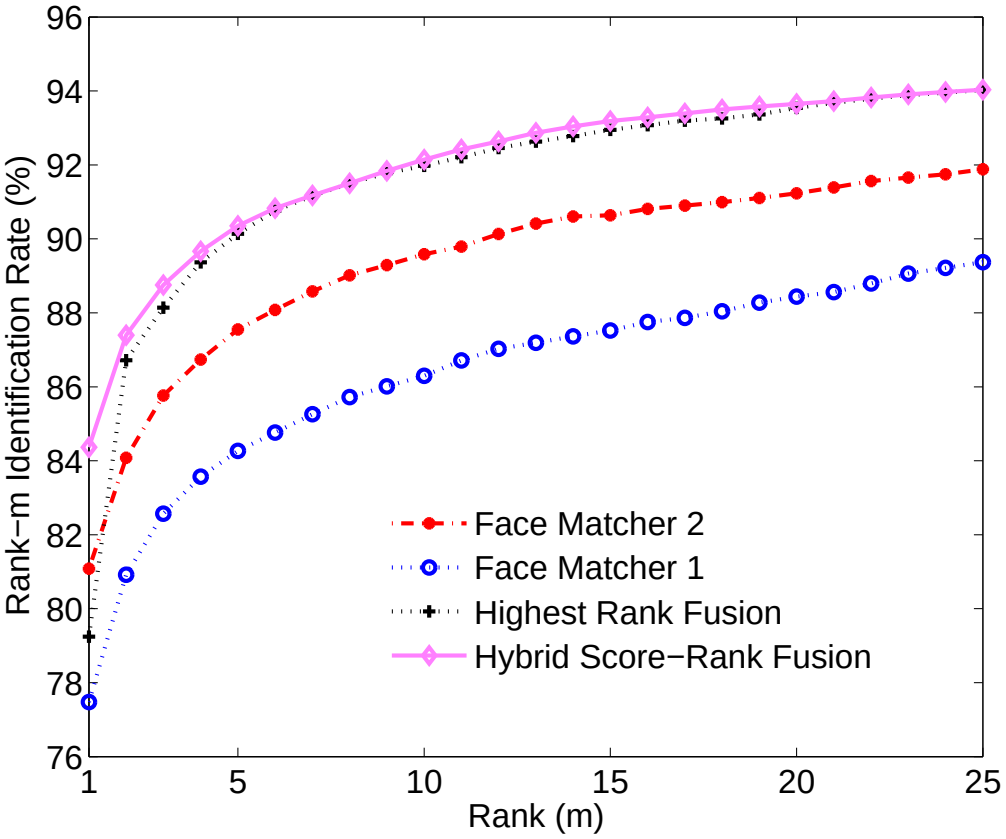



Figure 4.3: Cumulative Match Characteristic (CMC) curve of highest rank fusion and the hybrid score-rank fusion rules on the NIST-Face database ($K = 2, N = 3,000$).

face matcher and 84.1% for the score-rank fusion rule) due to the strong correlation between the two face matchers.

The results also indicate that the performance of the simplest rank level fusion scheme, namely, the highest rank method, is quite comparable to performance of the more complex score and rank fusion strategies for ranks greater than or equal to $K$, where $K$ is the number of matchers. Therefore, in practical multibiometric identification systems with a large number of users, it may be sufficient to use the highest rank method if the goal is to retrieve the top few matches. However, if the best rank-1 accuracy is desired and if the match score information is available, then the hybrid score-rank fusion rule can be employed.

## 4.4 Summary

While fusion in a multibiometric identification system is a more challenging problem due to the presence of large number of classes, we have shown that the likelihood ratio based fusion framework developed for a verification system can also be used for identification, provided the match scores of different users are assumed to be independent and identically distributed. We also proposed a scheme for rank level fusion in multibiometric identification that is based on converting the ranks into posterior probabilities. Furthermore, the rank posteriors can be directly combined with the posteriors obtained from the match score distributions to obtain a hybrid score-rank fusion rule. Finally, we have demonstrated that the proposed hybrid fusion rule consistently achieves high recognition rates at all ranks.

# Chapter 5

# Multibiometric Template Security

One of the most potentially damaging attacks on a biometric system is against the biometric templates. Attacks on the template can lead to the following four vulnerabilities: (i) A template can be replaced by an impostors's template to gain unauthorized access, (ii) a physical spoof can be created from the template (see [3,21,171]) to gain unauthorized access to the system (as well as other systems that use the same biometric trait), (iii) the stolen template can be replayed to the matcher to gain unauthorized access, and (iv) the templates can be used for cross-matching across different databases to covertly track a person without his/her consent. Due to these reasons, biometric templates (or the raw biometric images) should not be stored in plaintext form and fool-proof techniques are required to securely store the templates such that both the security of the application and the users' privacy are not compromised by adversary attacks. As shown in Chapters 3 and 4, multibiometric systems that fuse evidence from multiple biometric sources can provide significant improvement in the recognition accuracy. However, a multibiometric system requires storage of multiple

templates for the same user corresponding to the different biometric sources. Hence, template security is even more critical in multibiometric systems where it is essential to secure multiple templates of a user.

Although a number of approaches such as feature transformation and biometric cryptosystems have been proposed to secure templates [199], these approaches have been proposed primarily to secure a single template. While it is possible to apply these template protection schemes to each individual template separately, such an approach is not optimal in terms of security. The following simple analogy illustrates why securing the individual templates separately is not the best approach. Consider an application that requires the user to enter two separate 4-digit personal identification numbers (PIN) that are verified independently to provide access. An adversary attempting to break such a system would require at most $10^4$ attempts to guess each PIN. Since the PINs are verified independently, the maximum number of attempts needed to circumvent the system is only $2 \times 10^4$. On the other hand, if the application employs a single 8-digit PIN, the attacker would now need a maximum of $10^8$ attempts to circumvent the system, which would require more effort than cracking two 4-digit PINs. Protecting the individual templates separately is equivalent to having a scheme requiring multiple smaller PINs, which is less secure than a scheme that stores the multiple templates as a single entity (analogous to single large PIN).

In this chapter, we propose a unified scheme to secure multiple templates of a user in a multibiometric system by (i) transforming features from different biometric sources (e.g., fingerprint minutiae and iriscodes) into a common representation, (ii) performing feature-level fusion to derive a single multibiometric template, and (iii)

securing the multibiometric template using a single fuzzy vault construct [102]. We show that the proposed multibiometric template protection scheme has higher security and better recognition performance compared to the case where the individual templates are secured separately. We have developed a fully automatic implementation of a multibiometric fuzzy vault that can handle the following scenarios (i) multiple samples (e.g., two impressions from the same finger), (ii) multiple instances (e.g., left and right index fingers) and (iii) multiple traits (e.g., fingerprint and iris).

## 5.1 Review of Template Protection Schemes

Almost all the commercial biometric systems secure the stored templates by encrypting them using standard cryptographic techniques. Either a public key cryptosystem like RSA [115] or a symmetric key cipher like AES [1] is commonly used for template encryption. Since the above cryptosystems are generic, they can be directly applied to any biometric template and the encrypted templates are secure as long as the decryption key is secure. However, encryption is not a good solution for biometric template protection due to two main reasons. Firstly, encryption is not a smooth function and a small difference in the values of the feature sets extracted from the raw biometric data would lead to a very large difference in the resulting encrypted features. Recall that multiple acquisitions of the same biometric trait do not result in the same feature set (see Figure 1.3). Due to this reason, one cannot store a biometric template in an encrypted form and then perform matching in the encrypted domain. Hence, for every authentication attempt, (i) the template is decrypted, (ii) matching

is performed between the query and decrypted template and (iii) the decrypted template is then removed from memory. Thus, the template gets exposed during every authentication attempt. Secondly, the security of the encryption scheme depends on the decryption key. Hence, the decryption key needs to be securely stored in the system and if the key is compromised, the template is no longer secure. Because of these two reasons, standard encryption algorithms alone are not adequate for securing biometric templates and techniques that are designed to specifically account for the intra-user variability in the biometric data are needed.

The template protection schemes proposed in the literature can be broadly classified into two categories (see Figure 5.1), namely, *feature transformation* approach and *biometric cryptosystem*.

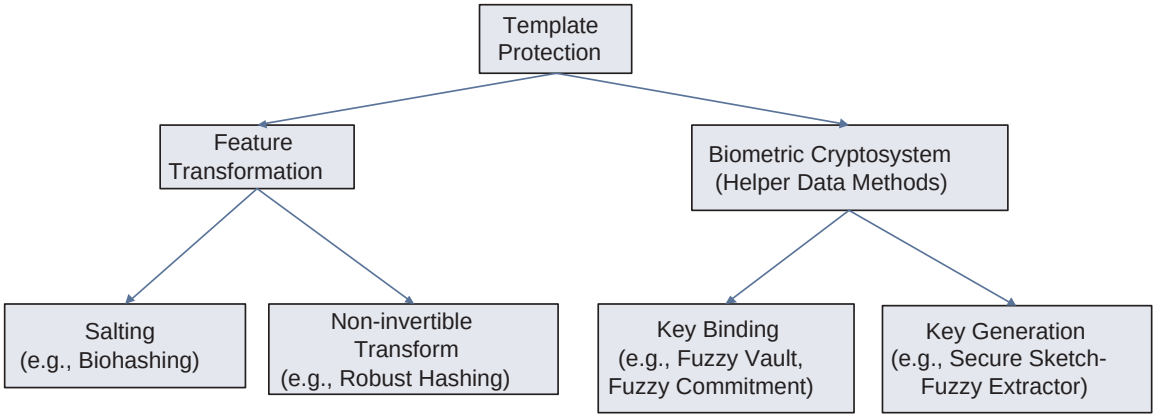



Figure 5.1: Categorization of template protection schemes.

### 5.1.1 Feature Transformation

In the feature transform approach, a transformation function ($\mathcal{F}$) is applied to the biometric template ($T$) and only the transformed template ($\mathcal{F}(T;K)$) is stored in

the database (see Figure 5.2). The parameters of the transformation function are typically derived from a random key ($K$) or a password. The same transformation function is applied to query features ($Q$) and the transformed query ($\mathcal{F}(Q;K)$) is directly matched against the transformed template ($\mathcal{F}(T;K)$). Depending on the characteristics of the transformation function $\mathcal{F}$, the feature transform schemes can be further categorized as *salting* or *non-invertible transforms*. In salting, $\mathcal{F}$ is invertible, i.e., if an adversary gains access to the key and the transformed template, she can recover the original biometric template (or a close approximation of it). Hence, the security of the salting scheme is based on the secrecy of the key or password. On the other hand, non-invertible transformation schemes typically apply a one-way function on the template and it is computationally hard to invert a transformed template even if the key is known.

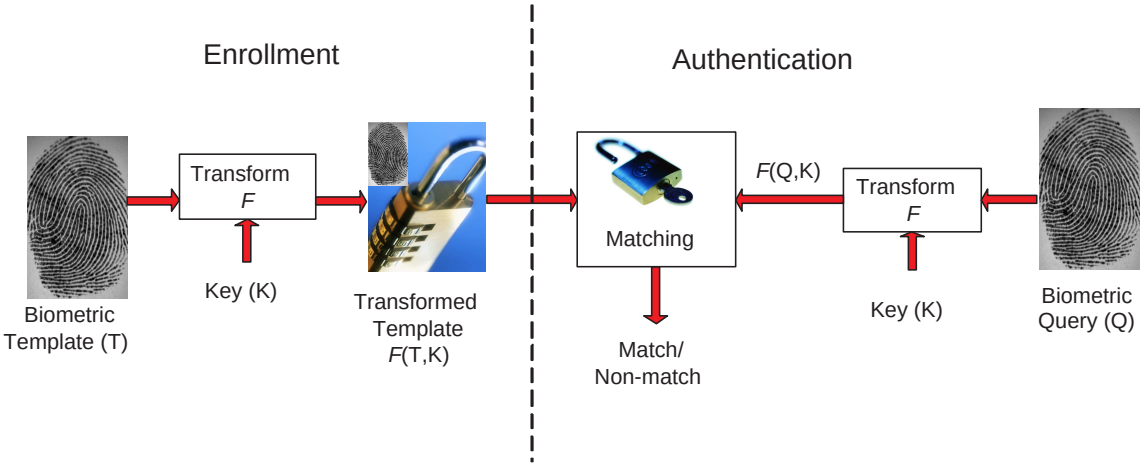



Figure 5.2: Authentication mechanism when the biometric template is protected using a feature transformation approach.

An example of salting approach is the random multi-space quantization technique proposed by Teoh et al. [187]. In this technique, the authors first extract the most dis-

criminative projections of the face template using the Fisher discriminant analysis [5] and then project the obtained vectors on a randomly selected set of orthogonal directions. This random projection defines the salting mechanism for the scheme. Similar biohashing schemes have been proposed for iris [38] and palmprint [43] modalities. Another example of salting is the cancelable face filter approach proposed in [174] where user-specific random kernels are convolved with the face images during enrollment and authentication. Non-invertible transformation functions have been proposed for fingerprint [158] and face [188] modalities in the literature.

### 5.1.2 Biometric Cryptosystems

Biometric cryptosystems [22, 198] were originally developed for the purpose of either securing a cryptographic key using biometric features or directly generating a cryptographic key from biometric features. However, they can also be used as a template protection mechanism. In a biometric cryptosystem, some public information about the biometric template is stored. This public information is usually referred to as *helper data* and hence, biometric cryptosystems are also known as helper data-based methods [199]. While the helper data does not (is not supposed to) reveal any significant information about the original biometric template, it is needed during matching to extract a cryptographic key from the query biometric features. Matching is performed indirectly by verifying the validity of the extracted key (see Figure 5.3). Error correction coding techniques are typically used to handle intra-user variations.

Biometric cryptosystems can be further classified as *key binding* or *key generation*

systems depending on how the helper data is obtained. When the helper data is obtained by binding a key (that is independent of the biometric features) with the biometric template, we refer to it as a *key-binding biometric cryptosystem*. Note that given only the helper data, it is computationally hard to recover either the key or the original template. Matching in a key binding system involves recovery of the key from the helper data using the query biometric features. If the helper data is derived only from the biometric template and the cryptographic key is directly generated from the helper data and the query biometric features, it leads to a *key generation biometric cryptosystem*.

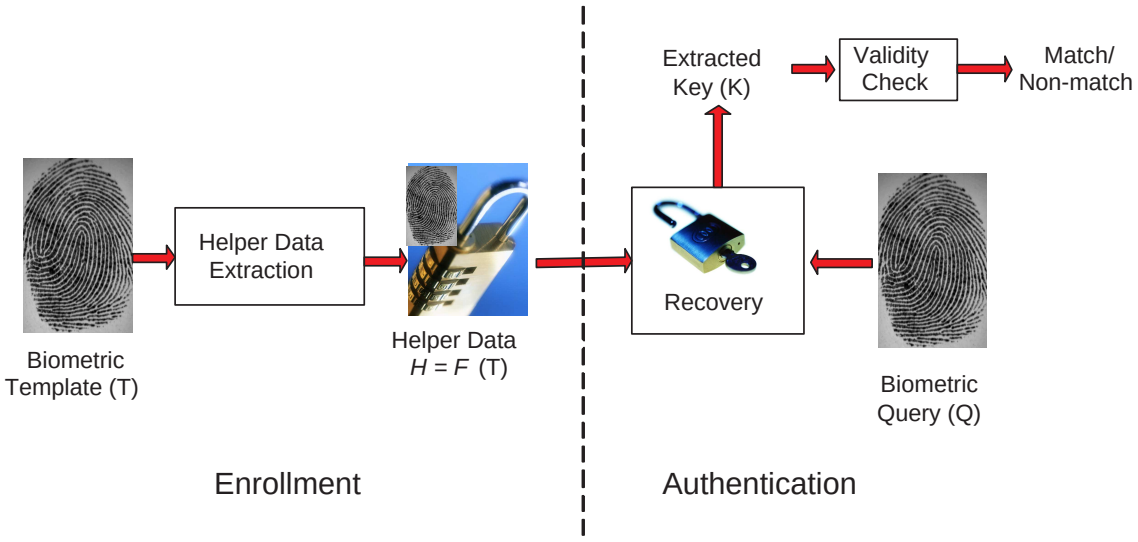



Figure 5.3: Authentication mechanism when the biometric template is secured using a key generation biometric cryptosystem. Authentication in a key-binding biometric cryptosystem is similar except that the helper data is a function of both the template and the key $K$, i.e., $H = \mathcal{F}(T; K)$.

A number of template protection techniques like fuzzy commitment [103], fuzzy vault [102], shielding functions [194] and distributed source coding [56] can be considered as key binding biometric cryptosystems. Other schemes for securing biometric

templates such as the ones proposed in [51, 76, 105, 137, 138] also fall under this category. The fuzzy vault scheme proposed by Juels and Sudan [102] has become one of the most popular approaches for biometric template protection and its implementations for fingerprint [41, 42, 141, 196, 209], face [62], iris [117] and signature [68] modalities have been proposed. Recently, multibiometric fuzzy vaults based on multiple fingers [210] and fingerprint and voice [19] have also been proposed.

Direct cryptographic key generation from biometrics is an attractive proposition, but it is a difficult problem because of the intra-user variability. Early biometric key generation schemes such as those by Chang et al. [28] and Veilhauer et al. [200] employed user-specific quantization schemes. Information on quantization boundaries is stored as helper data, which is used during authentication to account for intra-user variations. Dodis et al. [55] introduced the concepts of *secure sketch* and *fuzzy extractor* in the context of key generation from biometrics. The secure sketch can be considered as helper data that leaks only limited information about the template (measured in terms of entropy loss), but facilitates exact reconstruction of the template when presented with a query that is close to the template. The fuzzy extractor is a cryptographic primitive that generates a cryptographic key from the biometric features.

Dodis et al. [55] proposed secure sketches for three different distance metrics, namely, Hamming distance, set difference and edit distance. Li and Chang [120] introduced a two-level quantization based approach for obtaining secure sketches. Sutcu et al. [185] discussed the practical issues in secure sketch construction and proposed a secure sketch based on quantization for face biometric. The problem of

generating fuzzy extractors from continuous distributions was addressed by Buhan et al. in [16]. Secure sketch construction for other modalities such as fingerprints [4,23], 3D face [214] and multimodal systems (face and fingerprint) [186] have also been proposed. Protocols for secure authentication in remote applications [14, 17] have also been proposed based on the fuzzy extractor scheme.

Some template protection techniques make use of more than one basic approach (e.g., salting followed by key-binding). We refer to such techniques as *hybrid* schemes. Template protection schemes proposed in [13, 145, 183, 184] are examples of the hybrid approach. A brief summary of the various template protection approaches is presented in Table 5.1. Apart from salting, none of the other template protection schemes require any secret information (such as a key) that must be securely stored or presented during matching.

The template protection schemes described in Table 5.1 have their own advantages and limitations in terms of template security, computational cost, storage requirements, applicability to different kinds of biometric representations and ability to handle intra-class variations in biometric data [198]. In this thesis, we focus on a specific biometric cryptosystem known as fuzzy vault and present (i) a fully automatic implementation of a minutiae-based fingerprint fuzzy vault where high curvature points derived from the orientation field are used to align the template and query minutiae, (ii) an iris cryptosystem based on the fuzzy vault framework to secure iriscode templates, and (iii) a multibiometric vault framework to secure multiple templates of a user in a multibiometric system as a single entity.

Table 5.1: Summary of different template protection schemes. Here, $T$ represents the biometric template, $Q$ represents the query and $K$ is the key used to protect the template. In salting and non-invertible feature transform, $\mathcal{F}$ represents the transformation function and $\mathcal{M}$ represents the matcher that operates in the transformed domain. In biometric cryptosystems, $\mathcal{F}$ is the helper data extraction scheme and $\mathcal{M}$ is the error correction scheme that allows reconstruction of the key $K$.

| Approach | What imparts security to the template? | What entities are stored? | How are intra-user variations handled? |
|---|---|---|---|
| Salting | Secrecy of key $K$ | Public domain: Transformed template $\mathcal{F}(T;K)$<br>Secret: Key $K$ | Quantization and matching in transformed domain $\mathcal{M}(\mathcal{F}(T;K),\mathcal{F}(Q;K))$ |
| Non-invertible transform | Non-invertibility of the transformation function $\mathcal{F}$ | Public domain: Transformed template $\mathcal{F}(T;K)$, key $K$ | Matching in transformed domain $\mathcal{M}(\mathcal{F}(T;K),\mathcal{F}(Q;K))$ |
| Key-binding biometric cryptosystem | Level of security depends on the amount of information revealed by the helper data $H$ | Public domain: Helper Data $H=\mathcal{F}(T;K)$ | Error correction and user specific quantization $K=\mathcal{M}(\mathcal{F}(T;K),Q)$ |
| Key-generating biometric cryptosystem | Level of security depends on the amount of information revealed by the helper data $H$ | Public domain: Helper Data $H=\mathcal{F}(T)$ | Error correction and user specific quantization $K=\mathcal{M}(\mathcal{F}(T),Q)$ |

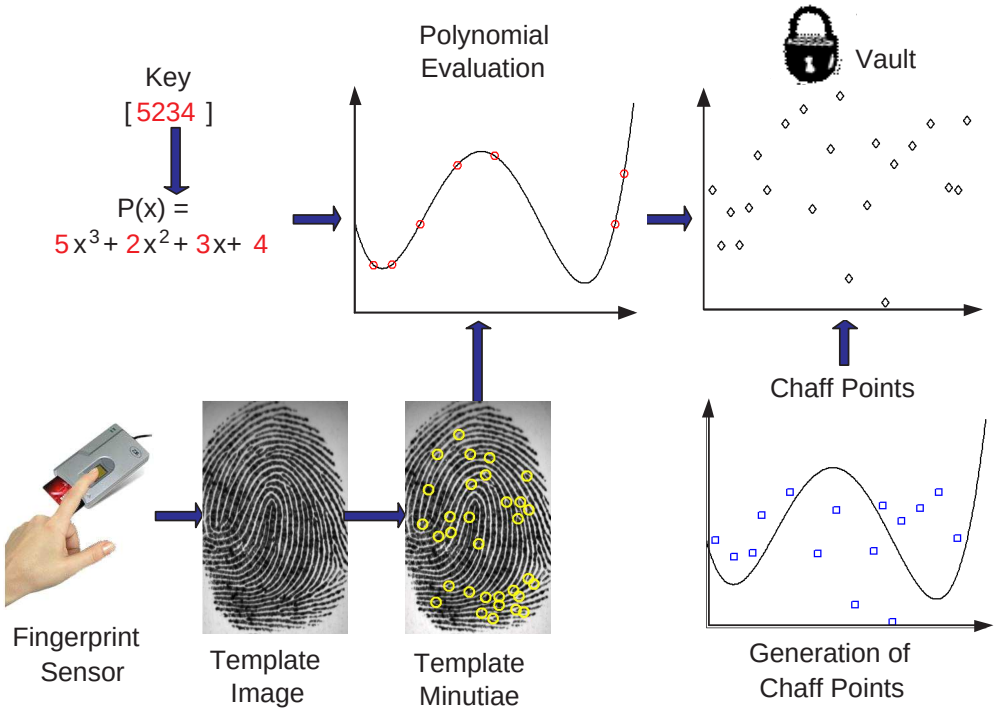
## 5.2 Fuzzy Vault

Fuzzy vault is a cryptographic construct that is designed to work with biometric features represented as an unordered set (e.g., minutiae in fingerprints). The security of the fuzzy vault scheme is based on the infeasibility of the polynomial reconstruction problem, which is a special case of the Reed-Solomon list decoding problem [11]. The fuzzy vault scheme works as follows (see Figure 5.4). Suppose that a user wishes to protect his biometric template, which is represented as an unordered set $X$, using a secret $K$ (e.g., a cryptographic key). Here, unordered set implies that all the elements in the set are unique and the order in which the elements of the set are listed is irrelevant. Note that this is true for minutiae representation of fingerprints. The user selects a polynomial $\mathcal{P}$ that encodes the secret $K$ in some way and evaluates the polynomial on all the elements in $X$. The user then chooses a large number of random chaff points that do not lie on the polynomial $\mathcal{P}$. The entire collection of points consisting of both points lying on $\mathcal{P}$ and those that do not lie on $\mathcal{P}$ constitute the vault $V$. The purpose of adding the chaff points is to conceal the points lying on $\mathcal{P}$ from an attacker. Since the points lying on $\mathcal{P}$ encode the complete information about the template $X$ and the secret $K$, concealing these points secures both the template and the secret simultaneously.

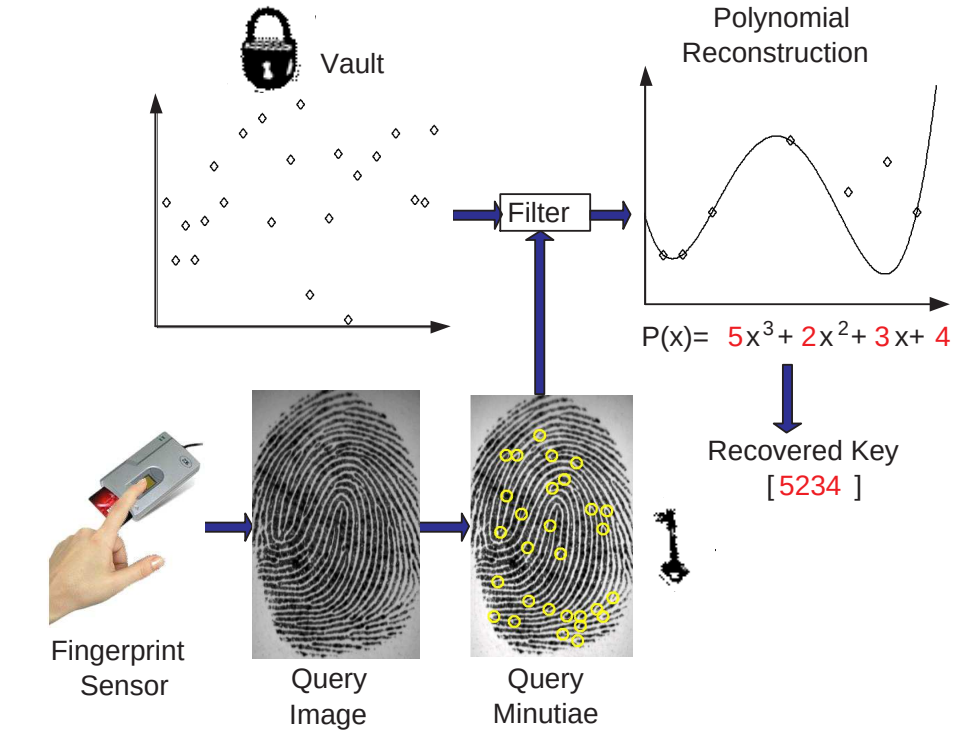
The user authentication based on the vault $V$ proceeds as follows. Let the query be represented as another unordered set $X'$. If $X'$ overlaps substantially with $X$, then the user can identify many points in $V$ that lie on the polynomial $\mathcal{P}$. If a sufficient number of points on $\mathcal{P}$ can be identified, an error correction scheme can be applied to

exactly reconstruct $\mathcal{P}$ and thereby decode the secret $K$. If a valid secret is decoded, the authentication is deemed to be successful. If $X^{'}$ does not overlap substantially with $X$, then it is infeasible to reconstruct $\mathcal{P}$ and the authentication is unsuccessful. Since the authentication can be successful even when $X$ and $X^{'}$ are not exactly the same, this scheme is referred to as a *fuzzy* vault.

The steps involved in creating the vault from the user's biometric template and the secret (vault encoding) are presented in Algorithm B.1 (see Appendix). All operations in this algorithm are carried out over a field $\mathcal{F}$. The algorithm has three parameters, namely, $n$, $r$ and $s$. Here, $r$ depends on the number of features that can be extracted from a user's biometric template (e.g., number of minutia points in the user's fingerprint). The parameter $s$ represents the number of chaff points that are added to the vault and this parameter influences the security of the fuzzy vault construction. If no chaff points are added, the vault leaks the information about the template and the secret. As more chaff points are added, the security of the vault increases. The degree of the polynomial, $n$, controls the tolerance of the system to errors in the biometric data during decoding. For example, $n$ determines the minimum number of matching minutiae required for successful vault decoding. A larger $n$ requires more number of minutiae matches. The function ENCODESECRET($K$) constructs a polynomial $\mathcal{P}$ of degree $n$ in variable $x$ such that $\mathcal{P}$ encodes the secret $K$ uniquely, i.e., given $\mathcal{P}$, we should be able to get back the secret $K$. A simple method to construct such a polynomial is to embed the secret in the coefficients of $\mathcal{P}$. The function PERMUTE($V^{'}$) randomly reorders the elements in $V^{'}$ to obtain the vault $V$.




(a)




(b)

Figure 5.4: Schematic diagram of the fuzzy vault scheme proposed by Juels and Sudan [102] based on fingerprint minutiae. (a) Vault encoding and (b) vault decoding.

Algorithm B.3 (see Appendix) presents the steps involved in retrieving the secret from the vault based on the user's biometric query (vault decoding). The output of this algorithm is either the secret $K$ or a value *null* indicating that the authentication is unsuccessful. The function RSDECODE($L^{'}$) is a $(r, n)$ Reed-Solomon decoding algorithm [7], which searches for a polynomial $\mathcal{P}$ of degree $n$ such that $\mathcal{P}\left(a_i^{'}\right) = b_i^{'}$ for more than $\frac{r+n}{2}$ values of $\left(a_i^{'}, b_i^{'}\right) \in L^{'}$. The RSDECODE function either outputs a polynomial $\mathcal{P}$ that satisfies the above conditions or a value *null* indicating that no such polynomial exists. The function DECODESECRET($p$) is the inverse of the ENCODESECRET function and it reconstructs the secret $K$ from the polynomial $\mathcal{P}$. The vault decoding algorithm successfully retrieves the secret $K$ if the number of errors (e.g., non-matching minutiae) in the biometric data $\left(|X - X^{'}|\right)$[1] is less than $\left(\frac{r-n}{2}\right)$. This ability to deal with intra-class variations in the biometric data, along with its ability to work with unordered sets, makes the fuzzy vault scheme a promising solution for biometric cryptosystems, particularly for fingerprints.

### 5.2.1 Fuzzy Vault Implementation

Since the introduction of the fuzzy vault scheme by Juels and Sudan, several researchers have attempted to implement it in practice for securing biometric templates. Clancy et al. [42] proposed a fuzzy vault scheme based on the location of minutia points (row and column indices in the image) in a fingerprint. They assumed that the template and query minutiae sets are pre-aligned, which is not a realistic

[1]The notation $|A|$ denotes the number of elements in a set $A$.

assumption in practical fingerprint authentication systems. Further, multiple (four) fingerprint impressions of a user were used during enrollment for identifying the reliable minutia points. The error correction step was simulated without being actually implemented. The False Reject Rate of their system was approximately 20-30% and they claimed that retrieving the secret was $2^{69}$ times more difficult for an attacker than for a genuine user.

The fingerprint-based fuzzy vault proposed by Yang et al. [209] also used only the location information about the minutia points. Four impressions were used during enrollment to identify a reference minutia, and the relative position of the remaining minutia points with respect to the reference minutia was represented in the polar coordinate system. This scheme was evaluated on a small database of 10 fingers and a FRR of 17% was reported. Chung et al. [41] proposed a geometric hashing technique to perform alignment in a minutiae-based fingerprint fuzzy vault. A modified fuzzy vault scheme was used for designing an asymmetric cryptosystem in [141]. Fuzzy vault implementations based on other biometric modalities such as face [62] and handwritten signature [68] have also been proposed.

Uludag et al. [197] introduced a modification to the fuzzy vault scheme, which eliminated the need for error correction coding. Uludag and Jain [196] also proposed the use of high curvature points derived from the fingerprint orientation field to automatically align the template and query minutiae sets. Our fingerprint-based fuzzy vault implementation [144] extends the ideas presented in [197] and [196] in order to achieve better performance on public-domain fingerprint databases.

## 5.3 Proposed Fingerprint-based Fuzzy Vault

We first propose a fuzzy vault implementation based on fingerprint minutiae. We use both the location and orientation attributes of a minutia point in our fuzzy vault implementation. These attributes are represented as a 3-tuple $(u, v, \theta)$, where $u$ indicates the row index in the image ($1 \leq u \leq U$), $v$ indicates the column index ($1 \leq v \leq V$) and $\theta$ represents the orientation of the minutia with respect to the horizontal axis ($1 \leq \theta \leq 360$). The algorithm presented in [87] is used for minutiae extraction.

We have implemented a modified version of the fuzzy vault construction that was proposed by Uludag et al. [197]. This modified fuzzy vault scheme does not require error correction coding. Instead, several candidate sets of size $(n+1)$ (where $n$ is the degree of the polynomial which encodes the secret) are generated from the unlocking set $L^{'}$ and polynomials are reconstructed using Lagrange interpolation. This method gives rise to several candidate secrets and Cyclic Redundancy Check (CRC) based error detection technique is used to identify the correct polynomial and hence decode the correct secret. The main advantage of this scheme is its increased tolerance to errors. Since only $(n+1)$ points are required to uniquely determine a polynomial of degree $n$, this scheme can retrieve the secret $K$ when the number of errors $\left(|X - X^{'}| = |L - L^{'}|\right)$ is less than $(r-n)$, i.e., it can tolerate twice the number of errors as the original fuzzy vault scheme. However, this method has a higher computational cost because it requires a large number of polynomial interpolations.

Our fingerprint-based fuzzy vault implementation differs from the implementation

in [197] and [196] in the following aspects.

1. In our implementation, we apply a *minutiae matcher* [87] *during decoding to account for non-linear distortion* in fingerprints whereas in [196], the minutia location information is coarsely quantized to compensate for distortion. Since deformation of the fingerprint ridges increases as we move away from the center of the fingerprint area towards the periphery, uniform quantization alone, as used in [196], is not sufficient to handle distortion. The minutiae matcher used in our implementation [87] employs an adaptive bounding box that accounts for distortion more effectively. This is one of the main reasons the proposed approach leads to a significant improvement in the genuine accept rate (GAR).

2. Only the location of minutia points was used for vault encoding in [196]. *We use both minutia location and orientation attributes*, which increases the number of chaff points that can be added because we can now add a chaff point whose location is close to a true minutia but with a different direction. Chang et al. [24] have shown that the number of possible chaff points affects the security of the vault. Hence, using both minutia location and orientation makes it more difficult for an attacker to decode the vault. At the same time, when a genuine user attempts to decode the vault, it is easier to filter out chaff points from the vault because it is less probable that a chaff point will match with a query minutia in both location and direction. This reduces the decoding complexity by eliminating most of the chaff points from the unlocking set.

3. *We use local image quality index estimated from the fingerprint in order to select*

*the most reliable minutiae* for vault encoding and decoding. In [197], minutiae selection was based on the value that is assigned to the minutiae in the field $\mathcal{F}$, which does not have any relation to the minutiae reliability. Our minutiae selection method is also more efficient than the one used in [42] where multiple fingerprint impressions were needed to determine reliable minutiae during encoding.

4. Although our alignment technique is similar to the one proposed in [196], *we have made significant changes to the curvature estimation and alignment steps* compared to [196], which results in a more accurate alignment between the template and query.

### 5.3.1 Vault Encoding

Figure 5.5 shows the block diagram of the proposed JS fuzzy vault encoding scheme. The field used for constructing the vault is $\mathcal{F} = GF(2^{16})$. We use the Galois field, $\mathcal{F} = GF(2^{16})$, for constructing the vault. The specific field $GF(2^{16})$ was chosen because it offers a sufficiently *large universe* (number of elements in the field) to ensure vault security [55] and is computationally convenient for the fuzzy vault application. The vault encoding process consists of the following eight steps.

1. Given the template fingerprint image $T$, we first obtain the template minutiae set $M^T = \{m_i^T\}_{i=1}^{N^T}$, where $N^T$ is the number of minutiae in $T$. The local quality index proposed in [32] is used to estimate the quality of each minutia in $T$. Let $q\left(m_i^T\right)$ be the quality of the $i^{th}$ minutia and $q^T = \{q\left(m_i^T\right)\}_{i=1}^{N^T}$

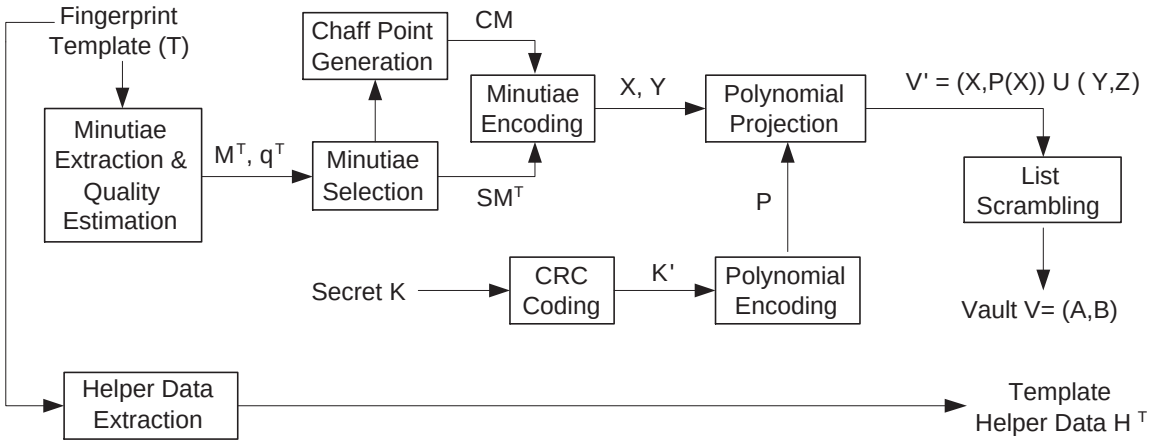



Figure 5.5: Proposed implementation of vault encoding.

be the quality set corresponding to minutiae set $M^T$. We also extract the high curvature points $H^T$ from the template image to be used for alignment during decoding. The details of extraction of high curvature points are presented in Section 5.3.3.

2. Since only $r$ genuine minutiae points are required to construct the vault, we apply a minutiae selection algorithm to the template minutiae set $M^T$. This selection algorithm first sorts the minutiae based on their quality and then sequentially selects the minutiae starting with the highest quality minutia. Moreover, the algorithm selects only well-separated minutiae, i.e., the minimum distance between any two selected minutia points is greater than a threshold $\delta_1$. The distance, $D_M$, between two minutia points $m_i$ and $m_j$ is defined as

$$D_M(m_i, m_j) = \sqrt{(u_i - u_j)^2 + (v_i - v_j)^2} + \beta_M \Delta(\theta_i, \theta_j), \qquad (5.1)$$

where $\Delta(\theta_i, \theta_j) = \min\Big(|\theta_i - \theta_j|, 360 - |\theta_i - \theta_j|\Big)$ and $\beta_M$ is the weight assigned to the orientation attribute (set to 0.2 in our experiments[2]). Selection of well-separated minutiae ensures that they are assigned unique values when they are encoded into the field $\mathcal{F}$. Let $SM^T = \{m_j^T\}_{j=1}^r$ denote the selected minutiae set. Note that if the number of minutia points in $T$ is less than $r$, or if the selection algorithm fails to find $r$ well-separated minutiae, we consider it as a failure to capture (FTC) error and no further processing takes place.

3. The chaff point set $CM = \{m_k^C\}_{k=1}^s$ is generated iteratively as follows. A chaff point $m = (u, v, \theta)$ is randomly chosen such that $u \in \{1, 2, \cdots, U\}$, $v \in \{1, 2, \cdots, V\}$ and $\theta \in \{1, 2, \cdots, 360\}$. The new chaff point is added to $CM$ if its minimum distance (as defined in equation (5.1)) to all the points in the set $SM^T \cup CM$ is greater than $\delta_1$.

4. The minutia attributes $u, v$, and $\theta$ are quantized and represented as bit strings of length $\mathcal{B}_u$, $\mathcal{B}_v$ and $\mathcal{B}_\theta$, respectively. If $\mathcal{B}_u$, $\mathcal{B}_v$ and $\mathcal{B}_\theta$ are chosen such that they add up to 16, we can obtain a 16-bit number by concatenating the bit strings corresponding to $u, v$, and $\theta$. Using this method, minutia points are encoded as elements in the field $\mathcal{F} = GF(2^{16})$. Let $X = \{x_j\}_{j=1}^r$ and $Y = \{y_k\}_{k=1}^s$ be the encoded values of selected template minutiae and chaff points, respectively, in the field $\mathcal{F}$.

[2]Since the variation in the orientation attribute of a minutia point is usually much larger compared to the variation in its location attribute, the orientation difference is assigned a smaller weight than the Euclidean distance between the minutiae locations. The specific value of 0.2 for $\beta_M$ was determined empirically as a tradeoff between eliminating as many chaff points as possible from the unlocking set while retaining as many genuine points as possible. The above tradeoff also determines the value of the threshold $\delta_2$ used in decoding.

5. Our scheme is designed to work with a secret key $K$ of length $16n$ bits, where $n$ is the degree of the encoding polynomial. We append a 16-bit CRC code to secret $K$ to obtain a new secret $K^{'}$ containing $16(n+1)$ bits. The generator polynomial $G(w) = w^{16} + w^{15} + w^2 + 1$, which is commonly known as IBM CRC-16, is used for generating the CRC bits.

6. The secret $K^{'}$ is encoded into a polynomial $\mathcal{P}$ of degree $n$ in $\mathcal{F}$ by partitioning it into $(n+1)$ 16-bit values $c_0, c_1, \cdots, c_n$ and considering them as coefficients of $\mathcal{P}$, i.e., $\mathcal{P}(x) = c_n x^n + \cdots + c_0$.

7. The polynomial $\mathcal{P}$ is evaluated at all the points in the selected minutiae set $X$ to obtain the set $\mathcal{P}(X) = \{\mathcal{P}(x_j)\}_{j=1}^{r}$. The corresponding elements of the sets $X$ and $\mathcal{P}(X)$ form the locking set $L = \{\Big(x_j, \mathcal{P}(x_j)\Big)\}_{j=1}^{r}$. A set $Z = \{z_k\}_{k=1}^{s}$ is obtained by randomly selecting values $z_k \in \mathcal{F}$ such that the points $(y_k, z_k)$ do not lie on the polynomial $\mathcal{P}$, i.e, $z_k \neq \mathcal{P}(y_k),\ \forall\ , k = 1, 2, \cdots, s$. The chaff set is defined as $C = \{(y_k, z_k)\}_{k=1}^{s}$. The union of locking and chaff sets is denoted as $V^{'}$.

8. The elements of $V^{'}$ are randomly reordered to obtain the vault $V$, which is represented as $V = \{(a_i, b_i)\}_{i=1}^{t}$, where $t = r + s$. Only the vault $V$ and the high curvature points $H^T$ are stored in the system.

### 5.3.2 Vault Decoding

The process of decoding the vault consists of the following steps (see Figure 5.6).

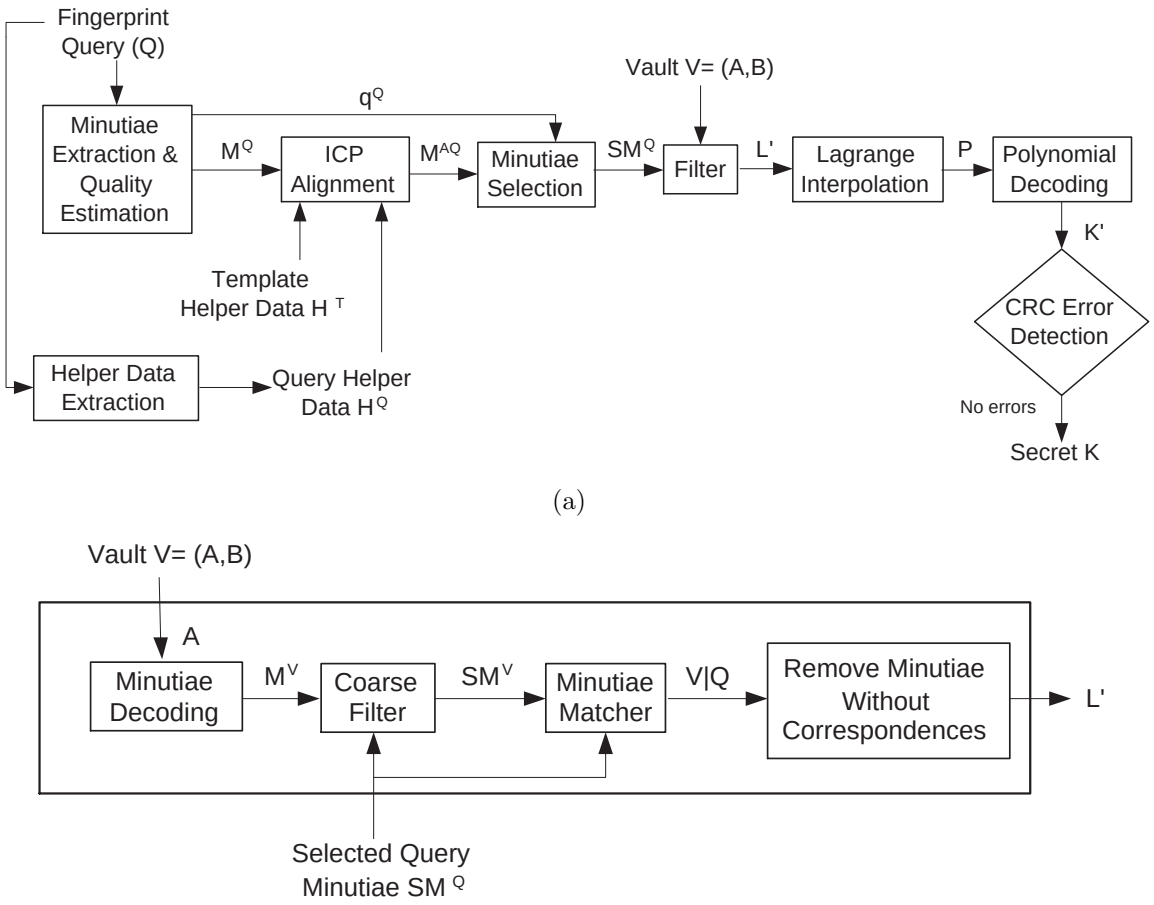



Figure 5.6: Proposed implementation of vault decoding. (a) Block diagram of the complete decoding process and (b) details of the filter used to eliminate the chaff points.

1. Given the query fingerprint image $Q$, we obtain the query minutiae set $M^Q = \{m_i^Q\}_{i=1}^{N^Q}$ and the high curvature points $H^Q$. The quality of each minutia in $Q$ is estimated and the quality set $q^Q = \{q\left(m_i^Q\right)\}_{i=1}^{N^Q}$ corresponding to $M^Q$ is obtained.

2. The alignment algorithm described in Section 5.3.3 is applied and the aligned query minutiae set $M^{AQ} = \{m_i^{AQ}\}_{i=1}^{N^Q}$ is obtained.

3. A minutiae selection algorithm is applied to select $r$ minutiae from the set $M^{AQ}$ based on their quality. The selected minutiae $SM^Q = \{m_j^Q\}_{j=1}^{r}$ are well-separated in the sense that the minimum distance (as defined in equation (5.1)) between any two selected minutiae is greater than $\delta_1$. If $N^Q < r$ or if the number of well-separated query minutiae is less than $r$, it is considered as failure to capture (FTC) and no further processing takes place.

4. The selected query minutiae are used to filter the chaff points in the vault as follows (see Figure 5.6(b)). The abscissa values of the points in the vault, i.e., $A = \{a_i\}_{i=1}^{t}$, are first represented as 16-bit strings. The 16-bit strings are partitioned into three strings of lengths $\mathcal{B}_u$, $\mathcal{B}_v$ and $\mathcal{B}_\theta$ which are then converted into quantized minutia attribute values $u$, $v$ and $\theta$. Thus, we obtain the set $M^V = \{m_i^V = (u_i, v_i, \theta_i)\}_{i=1}^{s}$.

5. The $i^{th}$ element of the set $M^V$ is marked as a chaff point if the minimum distance between the point $m_i^V \in M^V$ and all the selected minutiae in the query $m_j^Q \in SM^Q$ is greater than a threshold $\delta_2$. We refer to this process as a coarse

filter and it filters out a significant proportion of the chaff points (approximately 80%). Let $SM^V = \{m_k^V\}_{k=1}^{N^V}$ be a subset of $M^V$ containing only those elements that are not marked as chaff. Here, $N^V$ is the number of points in $M^V$ that are not marked as chaff and $N^V \ll s$. At this stage, a minutiae matcher [87] is applied to determine the corresponding pairs of minutiae in the sets $SM^V$ and $SM^Q$. Let $V|Q$ denote the set of correspondences and let $r^{'}$ be the number of correspondences. Since the size of the selected query minutiae set is $r$, we have $0 \leq r^{'} \leq r$ because each query minutiae can have no more than one corresponding minutia in $SM^V$. Note that it is also possible to directly apply the minutiae matcher to find correspondences between $M^V$ and $SM^Q$ without any coarse filtering. However, such a method is not effective because the presence of a large number of chaff points in the vault leads to a number of false correspondences. Hence, the coarse filter step is essential before the minutiae matcher is applied.

6. Only those elements of $V$ that are contained in $SM^V$ and which have a corresponding minutia in $SM^Q$ are added to the unlocking set $L^{'}$. The unlocking set is represented as $L^{'} = \{\left(a_i^{'}, b_i^{'}\right)\}_{i=1}^{r^{'}}$, where $\left(a_i^{'}, b_i^{'}\right) = \left(a_j, b_j\right)$ if $a_j$ has a corresponding minutia in $SM^Q$.

7. To find the coefficients of a polynomial of degree $n$, $(n+1)$ unique projections are necessary. If $r^{'} < (n+1)$, it results in authentication failure. Otherwise, we consider all possible subsets $L^{''}$ of size $(n+1)$ of the unlocking set $L^{'}$ and, for each subset, we construct a polynomial $\mathcal{P}^*$ by Lagrange interpolation. If

$L^{''} = \{\left(a_i^{'}, b_i^{'}\right)\}_{i=1}^{n+1}$ is a specific candidate set, $\mathcal{P}^*(x)$ is obtained as

$$\begin{aligned}\mathcal{P}^*(x) &= \frac{(x-a_2^{'})(x-a_3^{'})\cdots(x-a_{n+1}^{'})}{(a_1^{'}-a_2^{'})(a_1^{'}-a_3^{'})\cdots(a_1^{'}-a_{n+1}^{'})}\, b_1^{'} + \cdots \\ &\quad + \frac{(x-a_1^{'})(x-a_2^{'})\cdots(x-a_n^{'})}{(a_{n+1}^{'}-a_1^{'})(a_{n+1}^{'}-a_2^{'})\cdots(a_{n+1}^{'}-a_n^{'})}\, b_{n+1}^{'} \qquad (5.2)\end{aligned}$$

The above operations result in a polynomial $\mathcal{P}^*(x) = c_n^* x^n + c_{n-1}^* x^{n-1} + \cdots + c_0^*$.

8. The coefficients $c_0^*, c_1^*, \cdots, c_n^*$ of the polynomial $\mathcal{P}^*$ are 16-bit values which are concatenated to obtain a $16(n+1)$-bit string $K^*$ and CRC error detection is applied to $K^*$. If an error is detected, it indicates that an incorrect secret has been decoded and we repeat the same procedure for the next candidate set $L^{''}$. If no error is detected, it indicates that $K^* = K^{'}$ with very high probability. In this case, the 16-bit CRC code is removed from $K^*$ and the system outputs the secret $K$, which indicates a successful match.

### 5.3.3 Alignment based on High Curvature Points

The first step in matching two fingerprint images is to apply an alignment (registration) algorithm that can remove translation, rotation and possibly any non-linear distortion between the two images and determine the area of overlap. Although aligning two fingerprints is a difficult problem in any fingerprint authentication system, it is particularly more difficult in a biometric cryptosystem like fuzzy vault. This is

because the original fingerprint template is not available during authentication and only a transformed version of template is available in the vault.

Previous implementations of fingerprint-based fuzzy vault either assumed that the template and query fingerprint images are pre-aligned [42] or used a reference point (e.g., core point [161] or a reference minutia point [209]) for alignment. Though alignment based on a reference point is simple and computationally efficient, it is difficult to determine the reference point reliably. Even a small error in locating the reference point could lead to a false reject. To avoid this problem, Uludag and Jain [196] proposed the use of additional information derived from the fingerprint image to assist in alignment. While this additional data should carry sufficient information to accurately align the template and query images, it should not reveal any information about the template minutiae used for constructing the vault because any such leakage would compromise the security of the vault. Uludag and Jain derived the alignment data from the fingerprint orientation field. In particular, points of high curvature were used as the alignment data in [196] and an Iterative Closest Point (ICP) algorithm was used to determine the alignment between the template and the query based on this alignment data. Our proposed alignment scheme is similar to the one presented in [196] with some modifications.

### Extraction of High Curvature Points

An orientation field flow curve [47] is a set of piecewise linear segments whose tangent direction at each point is parallel to the orientation field direction at that point. Although flow curves are similar to fingerprint ridges, extraction of flow curves is

not affected by breaks and discontinuities, which are commonly encountered in ridge extraction. Points of maximum curvature in the flow curves along with their corresponding curvature values constitute the alignment data in our implementation. Therefore, the algorithm for extraction of high curvature points (see Figure 5.7) consists of four steps: (i) orientation field estimation, (ii) extraction of flow curves, (iii) determination of maximum curvature points and (iv) clustering of high curvature points.

Let $\mathcal{I}$ be a fingerprint image with $U$ rows and $V$ columns. A robust estimate of the orientation field for the given fingerprint image is obtained using the algorithm described in [46]. Let $\ell = (\lambda, \mu)$ be a point in $\mathcal{I}$, where $1 \leq \lambda \leq U$ and $1 \leq \mu \leq V$. Let $\phi_\ell$ be the orientation of the ridge flow with respect to the horizontal axis in the neighborhood of $\ell$. Let $O_\ell = (\cos\phi_\ell, \sin\phi_\ell)$ be the unit orientation vector at $\ell$. A flow curve with starting point $\ell_0 \in \mathcal{I}$ can be defined iteratively as

$$\ell_j = \ell_{j-1} + \rho \cdot \gamma \cdot O_{\ell_{j-1}}, \tag{5.3}$$

for $j = 1, 2, \cdots, J$. Here, $\rho = \{-1, +1\}$ defines the flow direction from $\ell_{j-1}$ to $\ell_j$, $\gamma$ is the length of the line segment from $\ell_{j-1}$ to $\ell_j$ and $O_{\ell_{j-1}}$ is the unit orientation vector at the point $\ell_{j-1}$. The process of tracing a flow curve is terminated when (i) the boundaries of the image are reached or (ii) $J$ exceeds a certain pre-defined threshold $J_{max}$. The parameter $\gamma$ determines the sampling interval of the flow curve and is set to 5 pixels in our experiments. Each starting point $\ell_0$ generates two curve segments $\left\{\ell_j^+\right\}_{j=1}^{J^+}$ and $\left\{\ell_j^-\right\}_{j=1}^{J^-}$ in opposite directions corresponding to $\rho = +1$

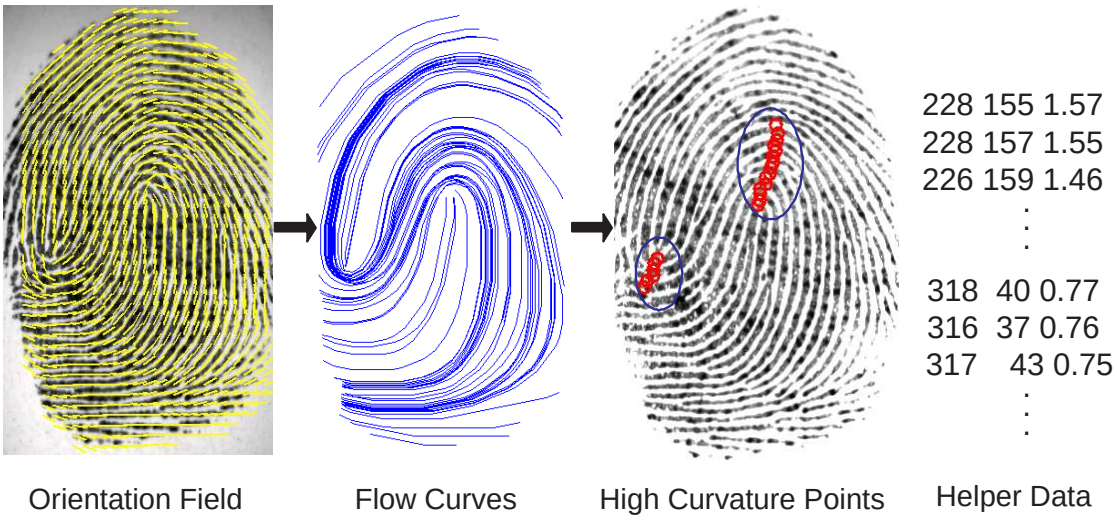


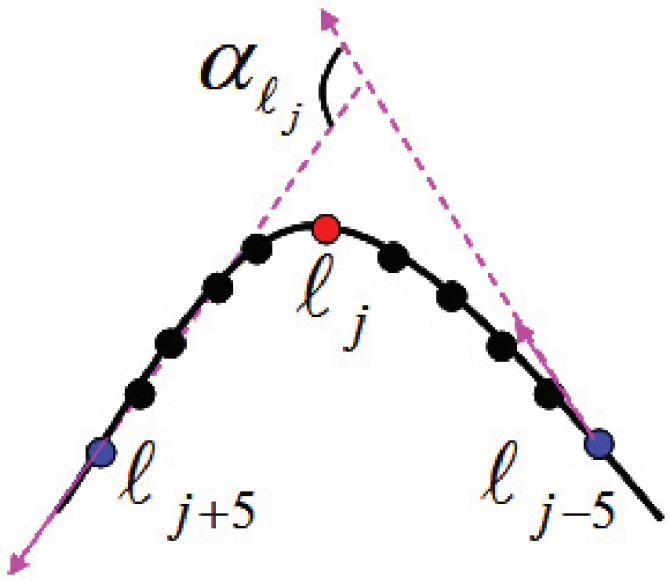

Figure 5.7: Algorithm for extraction of high curvature points.

and $\rho = -1$, respectively. The maximum number of samples in each curve segment, $J_{max}$, is set to 150. The two curve segments are then merged to get the complete flow curve, which is represented as a set of points $\left\{\ell_j\right\}_{j=1}^{J'}$, where $J' = J^+ + J^-$. By repeating this procedure with different starting points $\ell_0 \in \mathcal{I}$, we obtain a set of flow curves. Midpoints of the ridges in the thinned fingerprint image and points in whose neighborhood the orientation field changes significantly are chosen as the starting points.

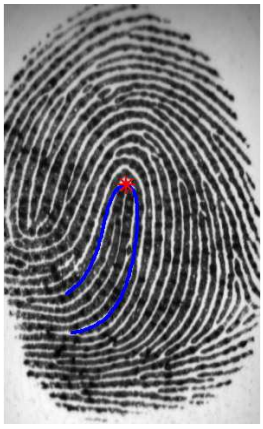
The curvature ($\omega$) of a point $\ell_j$ in a flow curve is defined as $\omega_{\ell_j} = 1 - \cos\alpha_{\ell_j}$, where $\alpha_{\ell_j}$ is the angle between the vectors that are tangent to the flow curve at the points $\ell_{j-\tau}$ and $\ell_{j+\tau}$, for all $\tau \leq j \leq J' - \tau$. The parameter $\tau$ is related to the sampling interval of the flow curve ($\gamma$) and is set to 5. The value of $\cos\alpha_{\ell_j}$ can be easily computed from the orientation field as $\cos\alpha_{\ell_j} = (\rho_{j-\tau} O_{\ell_{j-\tau}}) * (\rho_{j+\tau} O_{\ell_{j+\tau}})$, where $\rho_{j-\tau}$ is the flow direction from $\ell_{j-\tau}$ to $\ell_j$, $\rho_{j+\tau}$ is the flow direction from

$\ell_j$ to $\ell_{j+\tau}$ and $*$ indicates dot product. The value of $\omega_{\ell_j}$ is minimum (zero) when there is no change in direction as we go from $\ell_{j-\tau}$ to $\ell_{j+\tau}$ through $\ell_j$ and it attains its maximum value of 2 when the change in direction is $\pi$. For each flow curve, the curvature values for the points in the curve are estimated and local maxima in the curvature are detected. If the value of the local maximum is greater than a threshold $\sigma$ (set to 0.3), then the point is marked as a high curvature point and the 3-tuple $h = (\lambda, \mu, \omega)$, where $(\lambda, \mu)$ is the location and $\omega$ is the curvature value, is added to the alignment data set $H$. Figure 5.8 shows the procedure for curvature estimation at a point and a trace of the curvature values for a sample flow curve. The process of determining the maximum curvature points is repeated for all the flow curves, and the final alignment data set for the image $\mathcal{I}$ is obtained as $H^{\mathcal{I}} = \{h_i\}_{i=1}^{R^{\mathcal{I}}}$, where $R^{\mathcal{I}}$ is the number of high curvature points in $\mathcal{I}$. High curvature points usually tend to occur near the singular points (core and delta) in a fingerprint image. If the image has more than one singular point, the points in the alignment data set may have many clusters. Hence, a single-link clustering algorithm is applied to cluster the elements of the alignment data set based on the location of the points.

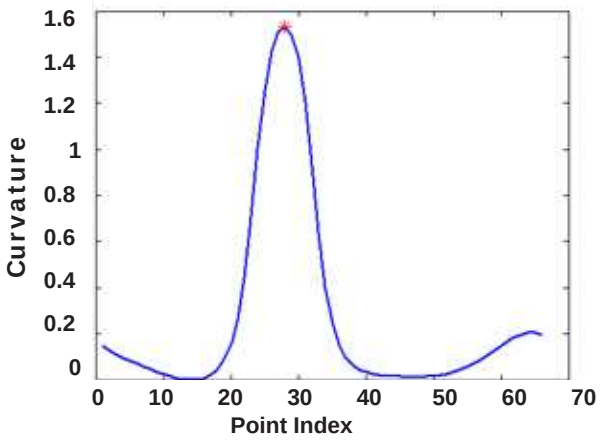
The proposed alignment data extraction scheme differs from the one proposed in [196] mainly in the definitions of curvature and local maxima in the curvature. The proposed definition of curvature leads to a smooth estimate of curvature with distinct local maxima. Further, unlike [196] where a single point having the maximum curvature is selected as the high curvature point, we apply a robust local maxima detection algorithm and utilize all the locally maximum points. This leads to better alignment data extraction for some types of fingerprint images such as whorls because




(a)



A Sample Flow Curve




Curvature Trace along the Flow Curve

(b)

Figure 5.8: Determination of maximum curvature points. (a) Curvature estimation at point $\ell_j$ and (b) trace of curvature for a sample flow curve along with the local maximum.

the flow curves near the core region of whorls generally tend to have more than one high curvature point (one above the core and one below the core).

**Alignment using ICP**

Let $T$ and $Q$ be the template and query fingerprint images, respectively. Let $H^T = \{h_i^T\}_{i=1}^{R^T}$ and $H^Q = \{h_j^Q\}_{j=1}^{R^Q}$ be the alignment data sets obtained from $T$ and $Q$, respectively. Let $M^Q = \{m_j^Q\}_{j=1}^{N^Q}$ be the query minutiae set, where $N^Q$ is the number of minutia points in $Q$. The goal of the alignment scheme is to find a rigid transformation $F$ that closely aligns $M^Q$ with the template minutiae set $M^T$ based on the alignment data sets $H^T$ and $H^Q$. Note that the template minutiae set $M^T$ is not available during alignment and only $H^T$ is known. We use the Iterative Closest Point (ICP) algorithm proposed by Besl and McKay [8] to align $H^T$ and $H^Q$ and estimate the rigid transformation $F$.

The ICP algorithm to align the template and query alignment data sets is shown in the appendix as Algorithm B.1. In this algorithm, the function INITTRANS$\left(H^T, H^Q\right)$ estimates an initial transformation between $H^T$ and $H^Q$ by aligning the center of mass of the points in $H^T$ and $H^Q$. The weighted distance, $D_H$, between two high curvature points $h_i$ and $h_j$ is defined as

$$D_H(h_i, h_j) = \sqrt{(\lambda_i - \lambda_j)^2 + (\mu_i - \mu_j)^2} + \beta_H|\omega_i - \omega_j|, \tag{5.4}$$

where $\beta_H$ weights the relative contribution of the Euclidean distance between the points (first term) and the difference in curvature (second term). The parameter

$\beta_H$ is set to 100 in our experiments. The function TRANS$\left(H^{T|Q}, H^Q\right)$ computes the transformation $F^{'}$ that minimizes the mean squared Euclidean distance between the locations of the corresponding points in $H^{T|Q}$ and $H^Q$. Algorithm B.1 is run until convergence or until a maximum number of iterations ($k_{max}$) is reached. The algorithm is said to converge if the change in the mean weighted distance ($MWD$) between the paired points is less than a threshold ($D_{stop}$). The values of $k_{max}$ and $D_{stop}$ are set to 200 and 0.01, respectively.

When the template and query images overlap only partially, it is possible that the overlap between the template and query alignment data sets is also partial. In such cases, all the high curvature points in the query may not have a corresponding point in the template. Algorithm B.1 strictly assigns a correspondence between every high curvature point in the query and the template, and this may lead to alignment errors when the overlap between the two sets is partial. To overcome this problem, we use the trimmed ICP algorithm [35], which basically ignores a proportion of the points in the query alignment data set whose distance to the corresponding points in the template alignment data set is large, i.e., we ignore the query points with large values of $D_H\left(h_j^{T|Q}, h_j^Q\right)$. The proportion of points to be ignored is found by minimizing an objective function (see [35] for details). The trimmed ICP algorithm is also robust to outliers in the alignment data sets.

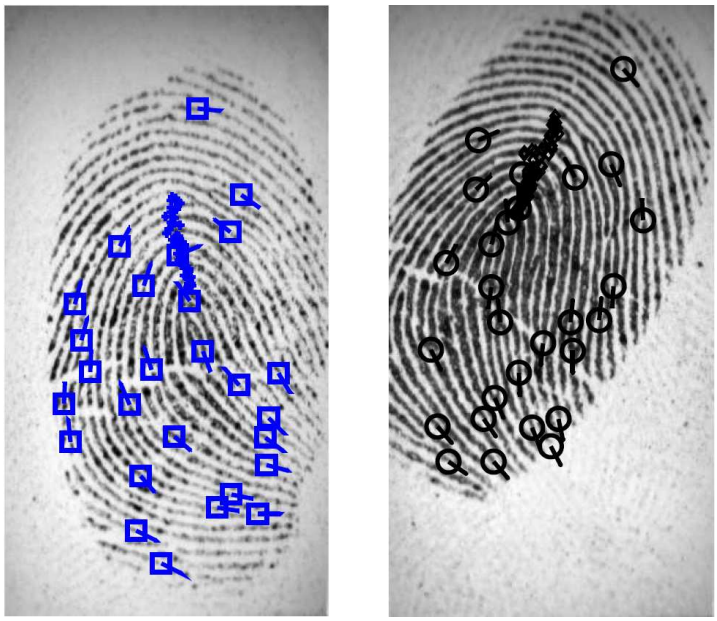
Based on the rigid transformation $F$ output by the ICP algorithm, we align the query minutiae set $M^Q$ with the template. Let $M^{AQ} = F\left(M^Q\right) = \{m_j^{AQ}\}_{j=1}^{N^Q}$ represent the query minutiae set after alignment. Figure 5.9 shows an example of

successful minutiae alignment based on high curvature points and trimmed ICP algorithm. In Algorithm B.1, it is assumed that both the alignment data sets $H^T$ and $H^Q$ have only a single cluster. If the number of clusters in $H^T$ and/or $H^Q$ is more than one, then the algorithm is repeated for all possible cluster pairs. In this scenario, there will be multiple aligned query minutiae sets. We select the aligned query minutiae set that gives the largest unlocking set $L'$.

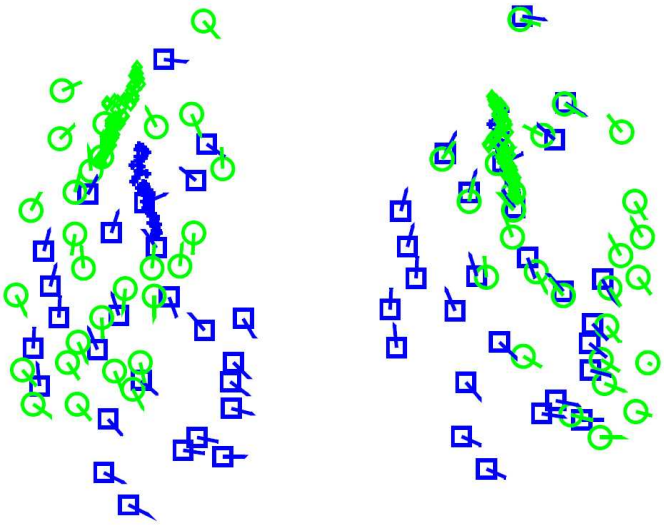
## 5.4 Proposed Iris Cryptosystem

The most common representation scheme used for matching iris images is the iriscode representation developed by Daugman [50]. The iriscode features are obtained by demodulating the iris pattern using quadrature 2D Gabor wavelets. In order to account for the variations in the pupil dilation, iris size and rotation, the rubber sheet model [50] is used to normalize the Gabor responses. The phase information in the resulting Gabor responses is then quantized into one of the four quadrants to produce a two-bit code for each local region. When the iris pattern is sampled at $R$ different radii and $S$ different angles, a $N$-bit iriscode sequence is generated, where $N = (2 \times R \times S)$. We use the algorithms described in [177] for pre-processing, segmentation and extraction of iriscodes from the iris images.

We now propose an iris cryptosystem to secure iriscode templates. Since the iriscode is a fixed length binary vector in which the relative order information between the bits is essential for matching, we cannot secure the iriscode directly using the fuzzy vault framework. To overcome this problem, we construct the iris cryptosystem in



(a) (b)



(c) (d)

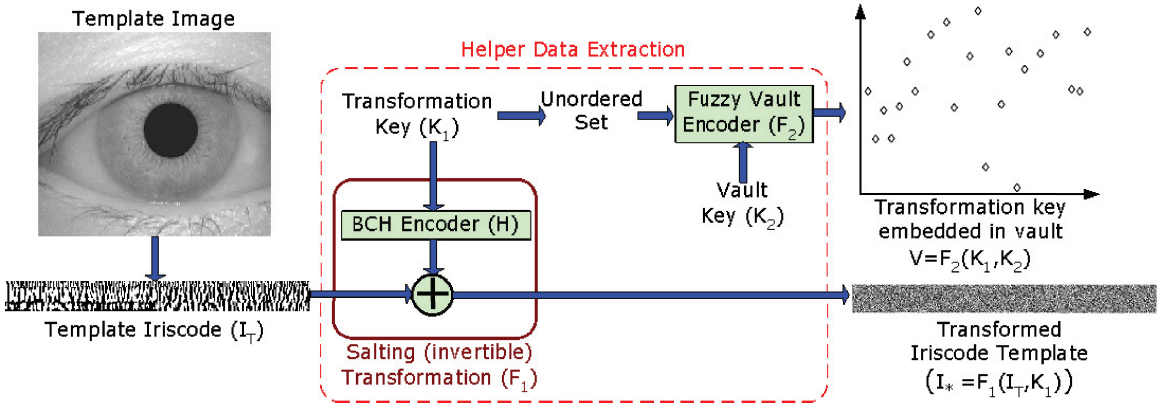
Figure 5.9: An example of successful minutiae alignment based on high curvature points and ICP algorithm. (a) Template image with minutiae and high curvature points, (b) query image with minutiae and high curvature points (c) template and overlaid query minutiae prior to alignment and (d) template and overlaid query minutiae after alignment. In this figure, the template minutiae are represented as squares (tails indicate the minutia direction) and the query minutiae are represented as circles. The template and query high curvature points are represented as asterisks and diamonds, respectively.

two steps (see Figure 5.10). In the first step, we apply a salting (invertible) transform to the iriscode template based on a randomly generated transformation key. Since the transformation is invertible, the security of the transformed iriscode template relies on the security of the transformation key. Hence, in the second step, we represent the transformation key as an unordered set and secure it using the fuzzy vault construct. Both the transformed iriscode template and the vault that embeds the transformation key constitute the helper data in this iris cryptosystem.

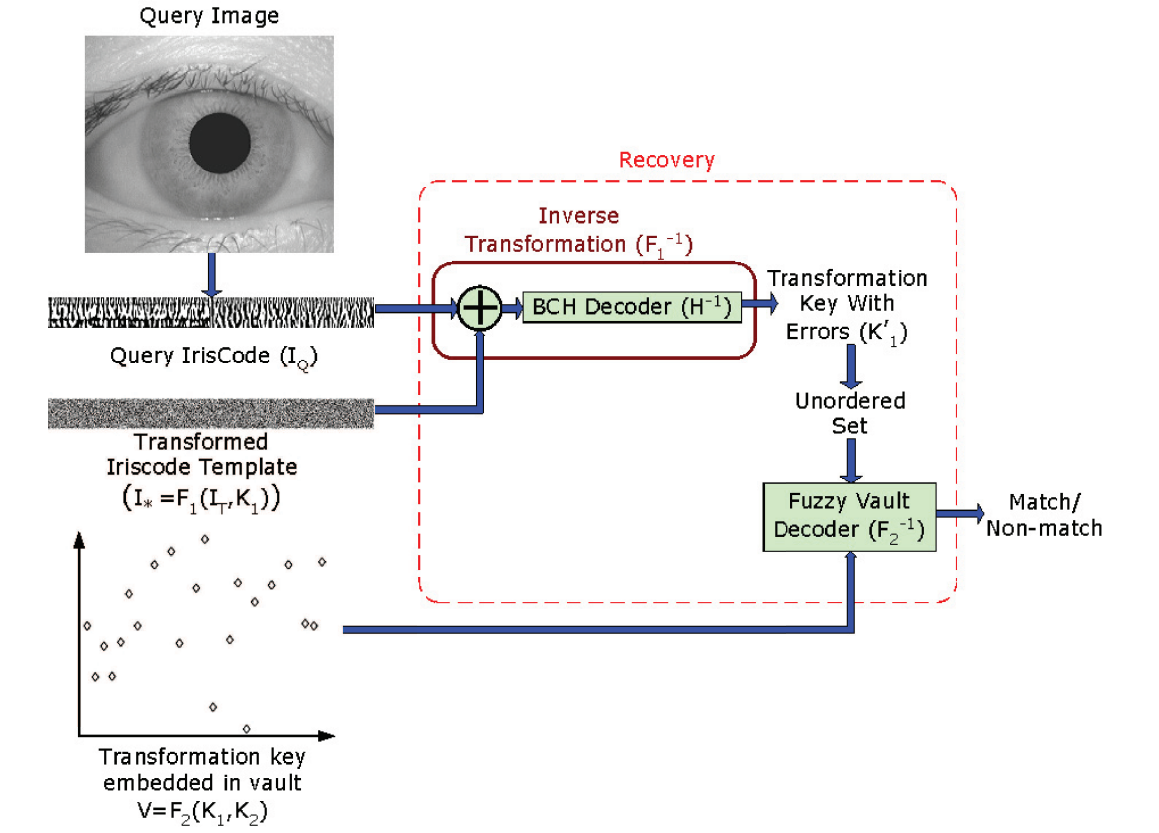
The proposed iris cryptosystem has two main advantages. Firstly, the salting step can be considered as a feature transformation function that converts a fixed length binary vector into an unordered set. This enables us to secure diverse biometric templates such as fingerprint minutiae and iriscode as a single multibiometric fuzzy vault. Moreover, both the salting and fuzzy vault steps can account for intra-user variations in the iriscode template. Due to the presence of two layers of error correction, the proposed iris cryptosystem allows larger intra-user variations in the iriscode template and hence, provides a high genuine accept rate.

### 5.4.1 Helper Data Extraction

The schematic diagram of helper data extraction scheme in the proposed iris cryptosystem is shown in Figure 5.10(a). The salting transform consists of two operations, namely, BCH encoding [122] and an exclusive-or operation. Let $H$ be a $(M_I, M_K)$ BCH encoding function, which takes a message $K$ of length $M_K$ ($M_K < M_I$) and appends $(M_I - M_K)$ error correcting symbols to it in order to generate a codeword




(a)




(b)

Figure 5.10: Schematic diagram of the iris cryptosystem based on iriscode features. (a) Enrollment or helper data extraction and (b) authentication or key recovery.

$I = H(K)$ of length $M_I$. In particular, we employ a primitive binary BCH encoding scheme, where $M_I$ is chosen to be $(2^m - 1)$ and $m$ is an integer greater than or equal to 3. The values of $M_I$ and $M_K$ determine the number of errors that can be corrected by the BCH coding scheme.

Let $I_T$ be a iriscode template of length $N_I$ bits that is to be secured using the fuzzy vault framework. First, we partition the template $I_T$ into $r$ non-overlapping components $\left[I_T^1, I_T^2, \cdots, I_T^r\right]$ such that each component $I_T^j$ $(j = 1, 2, \cdots, r)$ contains exactly $M_I$ bits. Here, $r$ is selected such that $rM_I \geq N_I$. When $N_I < rM_I$, appropriate number (i.e., $(rM_I - N_I)$) of zero bits are appended to the iriscode template $I_T$ to obtain the components $\left[I_T^1, I_T^2, \cdots, I_T^r\right]$. Next, we randomly generate $r$ binary vectors $K^1, K^2, \cdots, K^r$ each of length $M_K$ bits. These $r$ random binary vectors together constitute the transformation key $K_1$ of length $rM_K$ bits, i.e., $K_1 = \left[K^1, K^2, \cdots, K^r\right]$. The BCH encoder $H$ is applied individually to the binary vectors $K^1, K^2, \cdots, K^r$ to obtain the codewords $H(K^1), H(K^2), \cdots, H(K^r)$. Note that $H(K^j), j = 1, 2, \cdots, r$ is a binary vector of length $M_I$. Finally, an exclusive-or operation is performed between the $r$ codewords generated by the BCH encoder and the corresponding components of the iriscode template to obtain the components of the transformed iriscode. The transformed iriscode template $I_*$ can be represented as $\left[I_*^1, I_*^2, \cdots, I_*^r\right]$, where the $j^{th}$ component $I_*^j$ is given by $I_*^j = I_T^j \oplus H(K^j)$ , $\oplus$ denotes the exclusive-or operation and $j = 1, 2, \cdots, r$. Hence, the complete salting transformation can be represented as a function $F_1$ that takes the iriscode template $I_T$ and the transformation key $K_1$ as inputs and generates the transformed iriscode $I_*$ such that $I_* = F_1(I_T, K_1)$.

The transformation key $K_1$ is secured using the fuzzy vault construct as follows. Since the value of $M_K$ is set to 16 in our implementation, we can directly represent the $r$ components of the transformation key as elements in the Galois field $GF(2^{16})$. Our authentication (or key recovery) scheme has been designed in such a way that it does not require the relative order information between the components of key $K_1$. Hence, the components of the transformation key $K_1$ can be directly represented as an unordered set $X = \{x_j\}_{j=1}^{r}$, where $x_j$ is the representation of the component $K^j$ in $GF(2^{16})$. Let $Y = \{y_k\}_{k=1}^{s}$ be the set of chaff points such that $y_k \in GF(2^{16}), y_k \neq x_j, \forall\ j = 1, 2, \cdots, r$ and $k = 1, 2, \cdots, s$. Based on these two sets $X$ and $Y$ and a different key $K_2$ (referred to as the vault key with size $16n$ bits), we can construct a fuzzy vault $V = \{(a_i, b_i)\}_{i=1}^{t}, t = r + s$ by following steps 5 through 8 in the vault encoding algorithm presented in section 5.3.1. As pointed out earlier, the transformed iriscode $I_*$ and the vault $V$ together constitute the helper data in the iris cryptosystem.

### 5.4.2 Authentication

The steps involved in authentication based on the proposed iris cryptosystem are shown in Figure 5.10(b). Authentication or key recovery consists of two main stages. First, the inverse salting transform is applied to the transformed iris code template $I_*$ using the query iriscode $I_Q$. This facilitates the recovery of the transformation key used in vault encoding. Since the template and query iriscodes will not be identical due to intra-user variations, the recovered transformation key $K_1^{'}$ may have some

errors. In the second step, the transformation key $K_1^{'}$ is used to decode the vault $V$. If the template and query iriscodes are sufficiently similar, the recovered key $K_1^{'}$ will be sufficiently similar to $K_1$ and hence, the vault can be successfully decoded.

The inverse salting transform again consists of two operations, an exclusive-or followed by BCH decoding. Let $I_Q$ be the query iriscode of length $N_I$ bits. Similar to the encoding stage, we partition the query $I_Q$ into $r$ non-overlapping components $\left[I_Q^1, I_Q^2, \cdots, I_Q^r\right]$ such that each component $I_Q^j$ $(j = 1, 2, \cdots, r)$ contains exactly $M_I$ bits. An exclusive-or operation is performed between the $r$ components of the query iriscode and the corresponding components of the transformed iriscode to obtain the corrupted codewords. The $j^{th}$ corrupted codeword, $H^{'}(K^j)$, is given by $H^{'}(K^j) = I_*^j \oplus I_Q^j = I_Q^j \oplus I_T^j \oplus H(K^j) = e^j \oplus H(K^j)$, where $e^j$ is error vector indicating the differences between $I_Q^j$ and $I_T^j$ for $j = 1, 2, \cdots, r$. Let $H^{-1}$ be a $(M_I, M_K)$ primitive binary BCH decoding function that takes a corrupted codeword $H^{'}(K)$ of length $M_I$ and decodes it into a message $K^{'}$ of length $M_K$. If the Hamming distance between the corrupted codeword $H^{'}(K)$ and the original codeword $H(K)$ is less than the error correcting capability of the BCH coding scheme, then the decoded message $K^{'}$ would be the same as the original message $K$.

The corrupted codewords $H^{'}(K^j), j = 1, 2, \cdots, r$ are decoded using the BCH decoder to recover the components of the transformation key $K^{'j}$. If there are limited number of bit differences between the template and query iriscode components, the BCH decoder can account for those variations and the corresponding components of the transformation key can be recovered without any errors. However, due

to problems such as occlusion, there may be large differences between some of the template and query iriscode components and the corresponding components of the transformation key cannot be recovered correctly. The components of the recovered transformation key are represented as an unordered set $X^{'} = \{x_j^{'}\}_{j=1}^{r}$, where $x_j^{'}$ is the representation of the component $K^{'j}$ in $GF(2^{16})$. The unlocking set $L^{'}$ can be obtained as $L^{'} = \{(a_i, b_i)\}_{i=1}^{r^{'}}$, where $(a_i, b_i) \in V$ and $a_i = x_j^{'}$, for some $j \in 1, 2, \cdots, r$ and $r^{'} \leq r$. Steps 7 and 8 of the vault decoding algorithm presented in section 5.3.2 are applied to recover the vault key $K_2$. Successful recovery of the vault key indicates a match between the template and query iriscodes.

## 5.5 Multibiometric Fuzzy Vault

In a multibiometric system, there are multiple templates for each user corresponding to the different biometric sources. We propose a feature-level fusion to derive a single multibiometric template from the individual templates and secure the multibiometric template using a single fuzzy vault construct. In particular, we show how the multibiometric template can be derived in the following three scenarios, (i) multiple impressions of the same finger, (ii) multiple instances (e.g., left and right index fingers) and (iii) multiple traits (e.g., fingerprint and iris).

When multiple fingerprint impressions of the same finger are available for vault encoding, we can apply a mosaicing technique [170] to combine the minutiae and high curvature points from the individual images into a single mosaiced template and alignment data set. When multiple impressions are available for decoding, we use

them sequentially to unlock the vault. The decoding is successful if at least one of the two queries succeeds in unlocking the vault.

When multiple instances of the same biometric trait are available for a user, we can obtain the multibiometric template by concatenating the different feature sets. For example, if $M_{F_1}^T$ and $M_{F_2}^T$ are the template minutiae sets derived from the right and left index fingers of a user, respectively, the combined minutiae set $M_F^T$ can be obtained as the union of the sets $M_{F_1}^T$ and $M_{F_2}^T$. The fuzzy vault for the combined minutiae set $M_F^T$ can be constructed using the same procedure described in section 5.3.1. The high curvature points from both the fingers are stored separately along with the single multibiometric vault. During authentication, the query and template minutiae sets of the two fingers are aligned independently. The aligned query minutiae sets of the right and left index fingers are used to filter the chaff points from the vault to generate two unlocking sets $L'_{F_1}$ and $L'_{F_2}$. Either the union or the largest unlocking set can be considered as the final unlocking set that is used for polynomial reconstruction.

Figure 5.11 shows the encoding phase of a multimodal fuzzy vault with fingerprint and iris modalities. In this scenario, a feature transformation function is applied to the iriscode template to convert it into an unordered set with the help of a transformation key. The salting transform described in section 5.4.1 can be used for this purpose. Let $X_F$ and $X_I$ be the set of feature points generated by the fingerprint and iris modalities, respectively. Note that all elements of the sets $X_F$ and $X_I$ are in Galois Field $GF(2^{16})$. The union, $X$, of the two sets $X_F$ and $X_I$ is formed such that the Hamming distance between any two elements in the union is greater than or

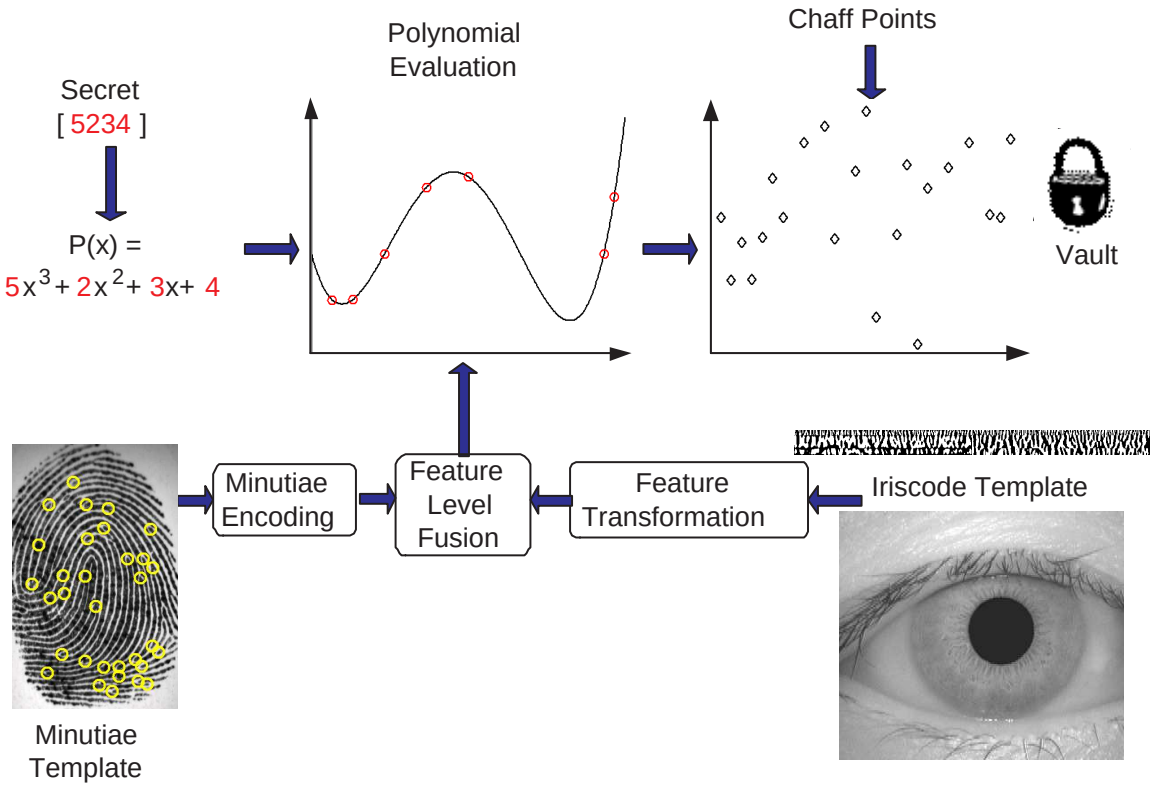



Figure 5.11: Schematic diagram of a multimodal (fingerprint and iris) fuzzy vault.

equal to 2. Here, the Hamming distance between any two elements in $GF(2^{16})$ is defined as the number of bit differences in the 16-bit binary representation of the elements. Steps 5 through 8 of the vault encoding algorithm presented in section 5.3.1 are then used for constructing the multimodal fuzzy vault. The high curvature points from the fingerprint and the transformed iriscode template are stored along with the vault as helper data. During authentication, the query iriscode is used to recover the transformation key from the transformed iriscode template. The aligned query minutiae set and the recovered iris transformation key are used to filter the chaff points from the vault and two unlocking sets $L_F^{'}$ and $L_I^{'}$ are generated. The union of the two unlocking sets is considered as the final unlocking set that is used for polynomial reconstruction.

## 5.6 Experimental Results

### 5.6.1 Fingerprint-based Vault

The performance of the proposed fingerprint-based fuzzy vault implementation has been evaluated on FVC2002-DB2 [128] and MSU-DBI [94] fingerprint databases (see Appendix A.2). We consider the following three scenarios for vault implementation.

1. One impression is used for encoding and one impression is used for decoding.

2. Two impressions are used for encoding and one impression is used for decoding.

3. Two impressions are used for encoding and two impressions are used for decoding.

The parameters used in our implementation for the two databases are listed in Table 5.2. The choice of polynomial degree ($n$) is related to the size of the secret to be secured. For example, if $n = 8$, we can secure a key of size 128-bits. Since the vault decoding is successful if $(n + 1)$ query minutiae match with the template minutiae, the parameter $n$ also affects the error rates. Since the number of minutiae varies for different users, using a fixed value of $r$ (the number of genuine minutiae points in the vault) across all users leads to a large failure to capture (FTC) rate. To overcome this problem, we fix the range of $r$ and determine its value individually for each user. The number of chaff points in the vault ($s$) is chosen to be approximately 10 times the number of genuine points in the vault, which is a reasonable tradeoff between the complexity of a brute force attack and storage requirements of the vault. The number of bits used for encoding the minutia attributes $u$, $v$ and $\theta$ into the field $\mathcal{F} = GF(2^{16})$ are $\mathcal{B}_u = 6$, $\mathcal{B}_v = 5$ and $\mathcal{B}_\theta = 5$, respectively. The allocation of bits determines the quantization step size for $u$, $v$ and $\theta$ and it depends on the image size. For the databases used here, the above parameter values for $\mathcal{B}_u$, $\mathcal{B}_v$ and $\mathcal{B}_\theta$ did not change the distribution of number of matching minutiae after quantization.

Table 5.2: Parameters used for fuzzy vault implementation.

| Parameter | FVC2002-DB2 | MSU-DBI |
|---|---|---|
| No. of genuine points in the vault, $r$ | 18-24 | 24-36 |
| Degree of encoding polynomial, $n$ | 7-10 | 10-12 |
| Total no. of points in the vault, $t$ | 224 | 336 |
| No. of chaff points in the vault, $s$ | 200-206 | 300-312 |
| Minimum distance between selected minutiae, $\delta_1$ | 25 | 25 |
| Maximum distance between a query minutia and points selected by the coarse filter, $\delta_2$ | 30 | 40 |

An example of successful vault operation for a user from FVC2002-DB2 when $n = 8$ is shown in Figure 5.12. Figure 5.12(f) shows that the ICP algorithm leads to correct alignment of query minutiae with the template minutiae concealed in the vault. The coarse filter and minutiae matcher eliminate most of the chaff points from the vault. The unlocking set mainly consists of genuine points from the vault. For example, in Figure 5.12(g) we observe that there is only one chaff point in the unlocking set. Since the number of genuine points in the unlocking set is more than 9, the decoding is successful in this example.

The criteria used for evaluating the performance of the vault are failure to capture rate (FTCR), genuine accept rate (GAR) and false accept rate (FAR). When the number of well-separated minutiae in the template and/or query fingerprint is less than $r$, it results in failure to capture. The genuine accept rate is defined as the percentage of attempts made by genuine users that resulted in successful authentication. Since a vault is constructed for each finger, the number of genuine attempts is 100 and 160 for the FVC and MSU databases, respectively. The false accept rate is the percentage of attempts made by impostors that resulted in successful decoding of a vault corresponding to a legitimate user. Impostor attempts were simulated by trying to decode a user's vault using impressions from all the other users. The number of impostor attempts is $9,900$ and $25,440$ for the FVC and MSU databases, respectively.

The first row of Table 5.3 shows the performance of the proposed vault implementation on the FVC2002-DB2 database for different key sizes when a single impression is used for encoding and decoding (impression 1 is used for encoding and impression 2 for decoding). For example, when the key size is 128 bits ($n = 8$), 91 out of 100 gen-

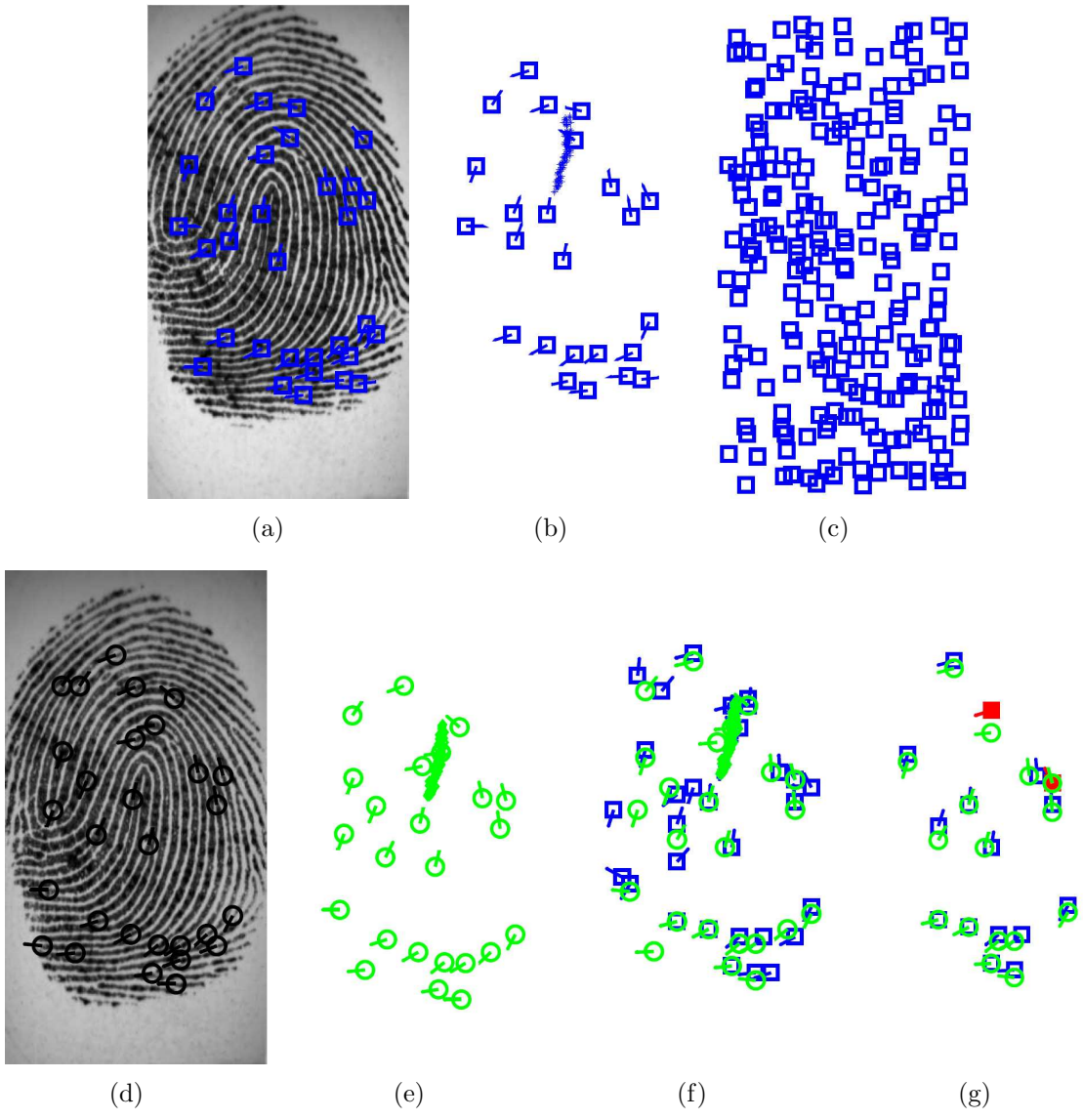



Figure 5.12: An example of successful operation of the fuzzy vault. (a) Template fingerprint image with minutiae, (b) selected template minutiae and high curvature points, (c) vault in which the selected template minutiae are hidden among chaff points (for clarity, minutiae directions are not shown), (d) query fingerprint image with minutiae, (e) selected query minutiae and high curvature points, (f) ICP alignment of template and query high curvature points and coarse filtering of chaff points, and (g) unlocking set obtained by applying a minutiae matcher that eliminates almost all the chaff points. The two points shown in filled squares in (g) are the only chaff points that remain in the unlocking set. Here, figures (a)-(c) represent vault encoding and (d)-(g) represent vault decoding.

uine attempts were successful. Among the 9 failed attempts, 2 were due to the lack of a sufficient number of minutiae in the template (FTC error). So, only 7 false rejects were actually encountered. For the same experiment, the fuzzy vault implemented in [196] was successful only in 61 out of 100 attempts with a FTCR of 16%. The high FTCR in [196] is due to errors in the extraction of high curvature points. If the alignment stage in the implementation of [196] is replaced with the one proposed in this thesis, the FTCR reduces to 2% and the GAR improves to 74%. This shows that the proposed alignment data extraction and alignment algorithms are more robust compared to those presented in [196]. The selection of reliable minutiae based on image quality and use of a minutiae matcher to account for non-linear distortion contribute to further improvement in the GAR from 74% to 91%. The net improvement in the GAR achieved by the proposed implementation over [196] is 30%.

In the case of MSU-DBI database, using single impressions for encoding and decoding results in a FTCR of 5.6% and a GAR of 82.5% for $n = 11$ (see the first row of Table 5.4). This decrease in performance compared to FVC2002-DB2 is due to the lower quality of images in the MSU database. However, the average number of matching minutiae in the MSU database is higher than in FVC2002-DB2, which allows us to accommodate a larger key size.

The proposed alignment technique based on high curvature points also performs better than registration based on core point. Since it is difficult to determine the core

Table 5.3: Performance of the proposed fingerprint-based fuzzy vault implementation on FVC2002-DB2 database. Here, $n$ denotes the degree of the encoding polynomial. The maximum key size that can be secured is $16n$ bits. The Failure to Capture Rate (FTCR), Genuine Accept Rate (GAR) and False Accept Rate (FAR) are expressed as percentages.

| **Scenario** | FTCR | $n = 7$ | | $n = 8$ | | $n = 10$ | |
|---|---|---|---|---|---|---|---|
| | | GAR | FAR | GAR | FAR | GAR | FAR |
| 1 Template, 1 Query | 2 | 91 | 0.13 | 91 | 0.01 | 86 | 0 |
| Mosaiced Template, 1 Query | 1 | 95 | 0.12 | 94 | 0.02 | 88 | 0 |
| Mosaiced Template, 2 Queries | 1 | 97 | 0.24 | 96 | 0.04 | 90 | 0 |

Table 5.4: Performance of the proposed fingerprint-based fuzzy vault implementation on MSU-DBI database. The Failure to Capture Rate (FTCR), Genuine Accept Rate (GAR) and False Accept Rate (FAR) are expressed as percentages.

| **Scenario** | FTCR | $n = 10$ | | $n = 11$ | | $n = 12$ | |
|---|---|---|---|---|---|---|---|
| | | GAR | FAR | GAR | FAR | GAR | FAR |
| 1 Template, 1 Query | 5.6 | 85 | 0.08 | 82.5 | 0.02 | 78.8 | 0 |
| Mosaiced Template, 1 Query | 2.5 | 88.1 | 0.09 | 83.1 | 0.02 | 81.2 | 0 |
| Mosaiced Template, 2 Queries | 0 | 96.9 | 0.16 | 92.5 | 0.03 | 87.5 | 0 |

point reliably, alignment based on core points leads to larger false rejects and failure to capture errors. For example, when core point based alignment[3] was used (instead of high curvature points) in the vault implementation, the FTCR increased from 2% to 6% in the FVC database and from 5.6% to 15.6% in the MSU database. The reasons for increase in FTCR are (i) no core point is present in some of the images (e.g., arch fingerprints) and (ii) the algorithm fails to find the core point in some images (e.g., images where the loops are not very prominent). These are well-known problems in core point detection. Furthermore, errors in finding the exact location and direction of the core point lead to a reduction in the GAR. The GAR decreases from 91% to 81% ($n = 8$) and from 82.5% to 77.5% ($n = 11$) for the FVC and MSU databases, respectively. These results clearly demonstrate the merits of using alignment based on high curvature points compared to core-based alignment.

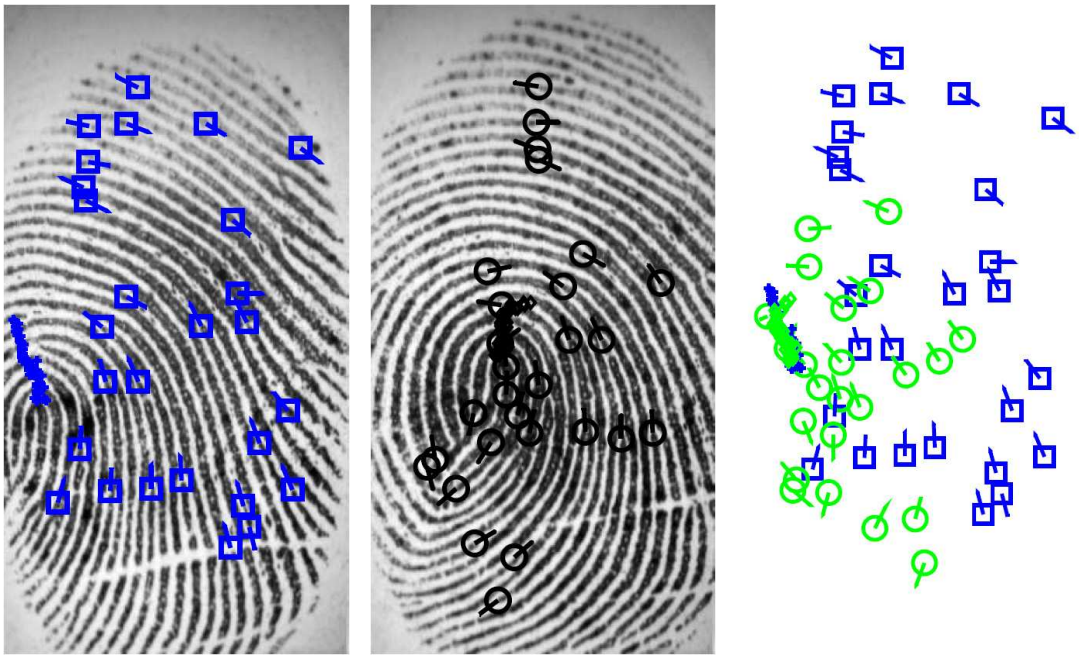
One way to improve the performance of the vault is to use multiple impressions (templates) from the same finger during enrollment. However, we cannot create a vault for each enrolled image because an attacker can compare the multiple vaults and identify the chaff points. Therefore, we obtain a single mosaiced template from two impressions and use the mosaiced minutiae to construct the vault. From row 2 of Table 5.3 we observe that mosaicing reduces the FTCR from 2% to 1% and also increases the GAR of the system for all values of $n$. The performance can be further improved by using multiple queries during authentication. In case of 128-bit key size ($n = 8$) for FVC2002-DB2 database, mosaiced template leads to a GAR of 94% and

[3]The core point was detected using the commercial Neurotechnologija Verifinger software, which was downloaded from `http://www.neurotechnologija.com`.

using two queries instead of one query increases the GAR to 96%. The use of multiple impressions also leads to a significant reduction in FTCR and increase in GAR for the MSU-DBI database (see rows 2 and 3 of Table 5.4).

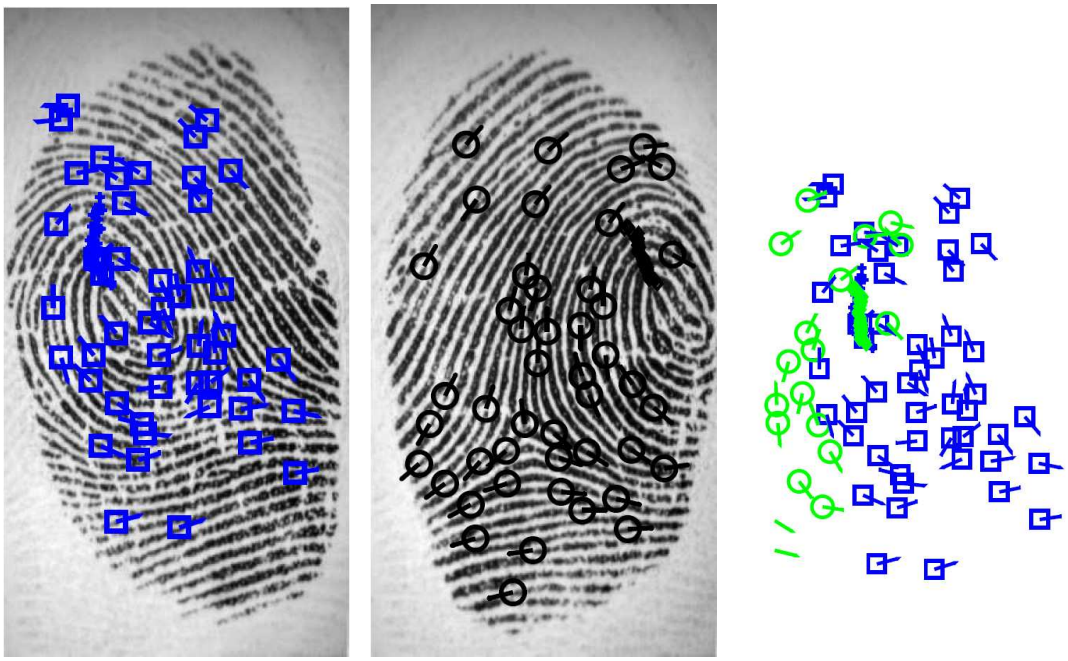
The false rejects in our experiments were either due to errors in alignment data extraction or due to insufficient number of matching minutiae in the overlapping region between the template and query. Figure 5.13 shows an example where the false reject is due to incorrect alignment data extraction. In this case, the high curvature points for template fingerprint are inaccurate because the region of high curvature (core region) is close to the image boundary (see Figure 5.13(a)). An example of failure due to insufficient number of overlapping minutiae is presented in Figure 5.14. While the alignment between the template and query images in Figure 5.14 is accurate, there are only 5 matching minutiae. This leads to a false reject because at least 9 genuine minutiae must be identified in the vault for successful decoding.

The FAR of the proposed fuzzy vault implementation is non-zero for smaller values of $n$. In the single impression scenario for FVC2002-DB2, when $n = 8$, we observed one false accept in $9,900$ impostor attempts. The template and query fingerprint pair that gives rise to a false accept is shown in Figure 5.15. Analysis of this false accept example indicates that there is indeed a set of 9 minutiae in the query that matches with the template minutiae in both location and direction (see Figure 5.15(c)). Since the vault decoding is successful if $(n + 1)$ points in the query minutiae set (of size $r$) match with the template minutiae, the genuine accept and false accept rates vary with $n$ when $r$ is fixed. Reducing $n$ increases both GAR and FAR and increasing $n$



(a) (b) (c)

Figure 5.13: Failure due to incorrect extraction of high curvature points. (a) Template fingerprint image with minutiae and high curvature points, (b) query fingerprint image with minutiae and high curvature points, and (c) ICP alignment of template and query high curvature points along with aligned template and query minutiae. High curvature points were incorrectly detected in the template because the high curvature region is near the boundary.



(a) (b) (c)

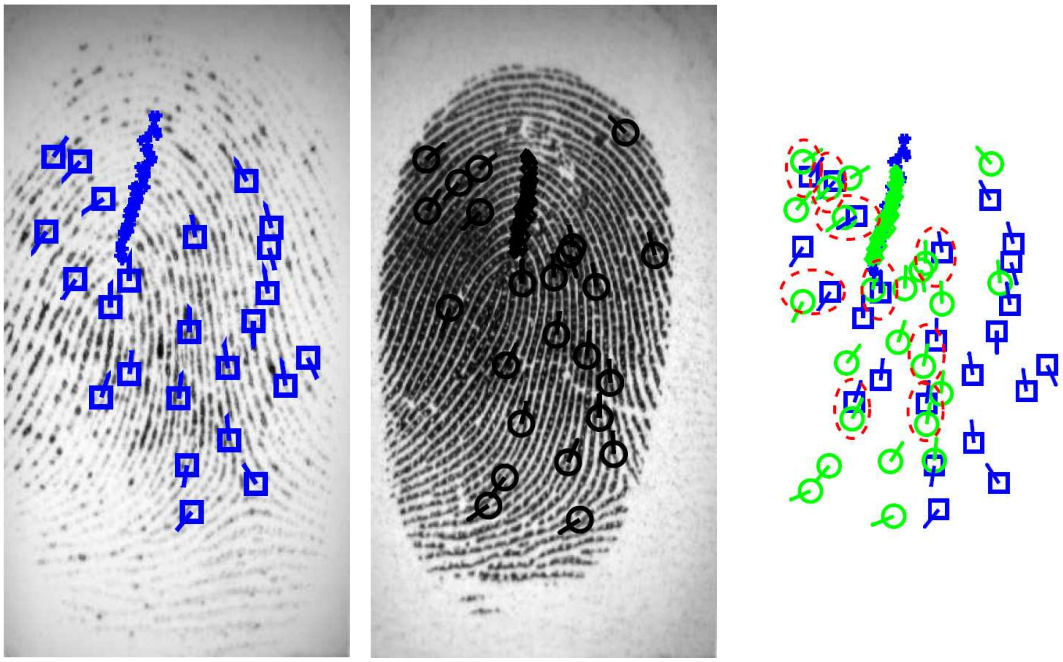
Figure 5.14: Failure due to partial overlap. (a) Template fingerprint image with minutiae and high curvature points, (b) query fingerprint image with minutiae and high curvature points, and (c) ICP alignment of template and query high curvature points along with aligned template and query minutiae. Though the alignment is accurate, there are only few matching minutiae in these two images.

lowers both GAR and FAR. As observed from Table 5.3, FAR is high when $n = 7$ and is zero when $n = 10$. We also observe a marginal decrease in the GAR when $n$ is increased from 7 to 10. The FAR for the MSU-DBI database also shows a similar behavior.

As pointed out earlier, a drawback of the modified fuzzy vault scheme in [197] compared to the original scheme in [102] is the need to verify multiple candidate secrets. In [197] it was reported that an average of 201 candidate secrets were evaluated corresponding to 52 seconds of computation in Matlab with a 3.4 GHz processor system. In our implementation, the use of minutiae orientation in addition to the minutiae location eliminates almost all the chaff points from the unlocking set. Therefore, the median number of candidate secrets that need to be evaluated is only 2 (mean is 33) and the median decoding time is 3 seconds (mean is 8 seconds) on a similar processor.

### 5.6.2 Iris Cryptosystem

The performance of the proposed iris cryptosystem has been evaluated on the CASIA iris database (see Appendix for details). In our implementation, the Gabor phase responses are sampled at $R(= 48)$ different radii and $S(= 360)$ angles to generate a $(48 \times 360 \times 2)$-bit iriscode. Further, we partition the iriscode into $r(= 48)$ components with each partition containing $M_I(= 1023)$ bits. We use a $(1023, 16)$ BCH coding scheme, which can correct up to 247 errors in a 1023-bit codeword. Thus, the BCH codes are capable of correcting approximately 25% of the errors in the query iriscode. The size of the transformation key $K_1$ used to secure the iriscode template is $(48 \times 16)$



(a) (b) (c)

Figure 5.15: An example of false accept when $n = 8$. (a) Template fingerprint image with minutiae and high curvature points, (b) query fingerprint image with minutiae and high curvature points, and (c) ICP alignment of template and query high curvature points along with aligned template and query minutiae. In (c), we observe that there are 9 matching minutiae between the query and the template (represented as dotted ellipses).

bits. The transformation key itself is secured using the fuzzy vault framework by using a vault key $K_2$ of size $16n$ bits, where $n$ is the degree of the polynomial used in vault encoding. We evaluate the performance of the iris cryptosystem at two different values of $n$ (10 and 11), which provide a false accept rate of less than 0.02%. The number of chaff points ($s$) used in the vault is set to 500.

Ideally, the bits in a query iriscode should directly correspond to the bits at the same location in the iriscode template. However, due to relative rotation of the iris pattern in the template and query iris images, the bits in a query iriscode may be shifted by a few locations with respect to the template iriscode. To account for this rotation offset, we cyclically shift the bits in the query iriscode by up to 3 locations both to the left and the right and repeat the authentication steps for each shifted query iriscode. A non-match decision is output only when none of the seven query iriscode patterns (one original and six shifted versions) are unable to recover the vault key.

The performance of the iris cryptosystem is shown in the first row of Table 5.6. The genuine accept rate of the iris cryptosystem is 88% at a false accept rate of less than 0.02%. The GAR of the Hamming distance-based iris matcher [50] that uses the original template and query iriscodes is approximately 94% at a FAR of 0.02%. Thus, there is a slight degradation in the GAR of the iris modality due to the application of the proposed template protection scheme. The reason for this degradation is that the BCH coding scheme has a strict threshold on the number of errors that can be corrected. When the number of bit differences between the template and query iriscode components is greater than 247, then the corresponding components of the

transformation key cannot be recovered. In some cases, features could not be reliably extracted from a relatively large region in the iris pattern due to factors like occlusion. The Hamming distance-based iris matcher accounts for this problem by determining the occluded regions (also known as the mask information) and ignoring the iriscode bits in those regions when computing the Hamming distance. However, the proposed cryptosystem cannot effectively handle this problem which leads to more false rejects.

### 5.6.3 Multibiometric Vault

The MSU-DBI fingerprint database [94] is used to evaluate the performance of the multifinger vault because it contains impressions from four different fingers (index and middle fingers) acquired from the same user. We use only the right and left index fingers in our experiments. The same parameters presented in the third column of Table 5.2 are used for constructing the vaults for the individual fingers. In the case of the multifinger vault, 48 to 72 genuine points are used in the vault and the total number of points in the vault ($t$) is set to 672. Thus, the number of chaff points in the vault is between 600 and 624. The performance of the multifinger vault is summarized in Table 5.5. When the largest of the two unlocking sets $L'_{F_1}$ and $L'_{F_2}$ is selected as the final unlocking set $L'_F$, the GAR improves significantly to 90% at a FAR of 0.02% compared to the single finger case. However, in this scenario there is no change in the size of the vault key ($K_2$) that determines the security of the vault. On the other hand, using the union of the two unlocking sets leads to a significant improvement in the security but leads to only a marginal improvement in the GAR.

Table 5.5: Performance of the multifinger (right and left index fingers) fuzzy vault on the MSU-DBI fingerprint database. The Failure to Capture Rate (FTCR), Genuine Accept Rate (GAR) and False Accept Rate (FAR) are expressed as percentages and the key size is expressed in bits.

| **Scenario** | FTCR | FAR = 0.02 | | FAR = 0 | |
|---|---|---|---|---|---|
| | | GAR | Vault Key Size | GAR | Vault Key Size |
| Right Index Finger | 5.6 | 82.5 | 176 | 78.8 | 192 |
| Left Index Finger | 8.8 | 75.6 | 176 | 69.4 | 192 |
| Both Fingers (Largest of the two unlocking sets) | 0 | 90 | 176 | 87.5 | 192 |
| Both Fingers (Union of the two unlocking sets) | 0 | 84.4 | 304 | 78.8 | 336 |

Finally, a virtual multimodal database derived from the MSU-DBI fingerprint and CASIA iris databases is used to evaluate the performance of a multimodal fuzzy vault that simultaneously secures the minutiae template from the right index finger and the iriscode template. The multimodal (right index finger and iris) database consists of 108 users obtained by randomly pairing the first 108 users in the MSU-DBI database with the users in the CASIA database. The number of genuine points in the multimodal vault is between 72 and 84 and the total number of points in the vault after adding the chaff points is 884. The third row in Table 5.6 shows the performance of the multimodal vault. The multimodal vault offers a significant improvement in the GAR compared to the individual modalities and also leads to higher security due to the larger key size.

Table 5.6: Performance of the multimodal (right index finger and iris) fuzzy vault on the virtual multimodal database derived from the MSU-DBI fingerprint and CASIA iris databases. The Failure to Capture Rate (FTCR), Genuine Accept Rate (GAR) and False Accept Rate (FAR) are expressed as percentage and the key size is expressed in bits.

| **Scenario** | FTCR | FAR = 0.02 | | FAR = 0 | |
|---|---|---|---|---|---|
| | | GAR | Vault Key Size | GAR | Vault Key Size |
| Iris | 0 | 88 | 160 | 88 | 176 |
| Right Index Finger | 5.6 | 82.5 | 176 | 78.8 | 192 |
| Right Index Finger + Iris (Union of the two unlocking sets) | 0 | 98.2 | 208 | 98.2 | 224 |

## 5.7 Security Analysis

Dodis et al. [55] defined the security of biometric cryptosystems in terms of the min-entropy of the helper data. Min-entropy of a random variable $A$ is defined as

$$H_\infty(A) = -\log\left(max_a P(A = a)\right). \tag{5.5}$$

Note that all the logarithms in this section are of base 2. Suppose the security of a system relies on the difficulty in guessing $A$. The best strategy for an adversary to circumvent this system is to start with the most likely value of $A$ and the min-entropy measures the security of the system in this scenario. Now consider a pair of random variables $A$ and $B$. Dodis et al. [55] defined the min-entropy of $A$ given $B$ as,

$$H_\infty(A|B) = -\log\left(max_a P(A = a|B = b)\right) \tag{5.6}$$

and the average min-entropy of $A$ given $B$ as

$$\tilde{H}_{\infty}(A|B) = -\log\left(E_{b\leftarrow B}\left[max_a P(A=a|B=b)\right]\right) = -\log\left(E_{b\leftarrow B}\left[2^{-H_{\infty}(A|B)}\right]\right). \tag{5.7}$$

We can analyze the security of the fuzzy vault framework by measuring the average min-entropy of the biometric template given the vault $V$.

### 5.7.1 Fingerprint-based Vault

Recall that the fingerprint-based vault $V = \{(a_i, b_i)\}_{i=1}^{t}$ is an unordered set of $t$ points consisting of $r$ points that lie on a polynomial $\mathcal{P}$ defined by the vault key $K$ and $s$ chaff points that do not lie on $\mathcal{P}$. Alternatively, if $X$ and $Y$ are the sets of genuine and chaff points, respectively, then $a_i \in X$ or $Y$, $\forall i = 1, 2, \cdots, t$. The vault can be decoded only if we can find a candidate set $L^{''} = \{(a_j, b_j)\}_{j=1}^{n+1}$, which is a subset of $V$ such that $a_j \in X$, $\forall (a_j, b_j) \in L^{''}$, where $n$ is the degree of the polynomial $\mathcal{P}$. If no other additional information is available, an adversary would have to decode the polynomial by randomly selecting subsets of $(n+1)$ points from $V$ and we refer to this case a brute-force attack.

Suppose that the adversary has knowledge of the fingerprint minutiae distribution model [215] and selects the candidate set $L^{''}$ based on this model. Let $L^{*} = \{(a_j, b_j)\}_{j=1}^{n+1}$ be the candidate set that is most likely to be selected based on the minutiae distribution model. Let $p_i$ be defined as the probability that $a_i$ corresponds to a genuine minutiae point, i.e., $p_i = P(a_i \in X)$, for $i = 1, 2, \cdots, t$ and $\sum_{i=1}^{t} p_i = 1$. If we know the distribution of location and orientation of minutiae in

a fingerprint, we can estimate $p_i$ for all the points in a given fingerprint-based vault $V$. Let us re-order the points in $V$ such that $p_i \geq p_{i+1},\ \forall i = 1, 2, \cdots, t-1$. If we sequentially select points from $V$ to form the candidate set based on the estimated $p_i$'s, then the probability of selecting the most likely genuine point is $p_1$, the probability of selecting the second most likely genuine point is $\frac{p_2}{(1-p_1)}$ and so on. Therefore, the probability that $L''$ takes the value $L^*$ is given by

$$P\left(L'' = L^*\right) \leq \frac{(n+1)! \prod_{i=1}^{n+1} p_i}{\prod_{i=1}^{n} \left(1 - \sum_{k=1}^{i} p_k\right)}. \tag{5.8}$$

Here, the factorial term is included because the candidate sets are unordered and the $(n+1)$ most likely elements from $V$ can be arranged in $(n+1)!$ ways to obtain $L^*$. Let $\mathcal{P}^*$ be the polynomial obtained by Lagrange interpolation of points in $L^*$. Since there are $\binom{r}{n+1}$ combinations of candidate sets $L''$ derived from $V$ that can decode the vault, the probability that $\mathcal{P}^*$ is the correct polynomial $\mathcal{P}$ is given by

$$P\left(\mathcal{P}^* = \mathcal{P}\right) \leq \frac{\binom{r}{n+1}(n+1)! \prod_{i=1}^{n+1} p_i}{\prod_{i=1}^{n} \left(1 - \sum_{k=1}^{i} p_k\right)}. \tag{5.9}$$

When $\mathcal{P}^*$ is equal to $\mathcal{P}$, the vault is decoded and the minutiae template $M^T$ is revealed. Therefore, the min-entropy of $M^T$ given $V$ can be computed as

$$H_\infty(M^T|V) \geq -log\left(\frac{\binom{r}{n+1}(n+1)! \prod_{i=1}^{n+1} p_i}{\prod_{i=1}^{n} \left(1 - \sum_{k=1}^{i} p_k\right)}\right). \tag{5.10}$$

If both the minutiae location and minutiae orientation are uniformly distributed, $p_i = 1/t,\ \forall i = 1, 2, \cdots, t$. In this case, the min-entropy of $M^T$ given $V$ can be

simplified as

$$
\begin{aligned}
H_{\infty}(M^T|V) &= -log\left(\frac{\binom{r}{n+1}(n+1)!\prod_{i=1}^{n+1}\frac{1}{t}}{\prod_{i=1}^{n}\left(1-\sum_{k=1}^{i}\frac{1}{t}\right)}\right) \\
&= -log\left(\frac{\binom{r}{n+1}(n+1)!\frac{1}{t^{n+1}}}{\prod_{i=1}^{n}\left(\frac{t-i}{t}\right)}\right) \\
&= -log\left(\frac{\binom{r}{n+1}(n+1)!}{t\prod_{i=1}^{n}(t-i)}\right) \\
&= -log\left(\frac{\binom{r}{n+1}(n+1)!(t-n-1)!}{t!}\right) \\
&= -log\left(\frac{\binom{r}{n+1}}{\binom{t}{n+1}}\right). \qquad (5.11)
\end{aligned}
$$

For example, if the size of the vault key is 160 bits (which corresponds to $n = 10$ in our implementation) and the number of genuine and chaff points in the vault are 30 and 300, respectively, under the assumption of uniform distribution of minutiae, the min-entropy of the fingerprint-based fuzzy vault is approximately 40 bits. Here, 40 bits of security implies that the expected number of candidate sets that need to be evaluated is $2^{40} \approx 2 \times 10^{12}$. This roughly corresponds to the same level of difficulty in guessing a 24-character ASCII password [18]. While a security of 40 bits may be considered as inadequate from the cryptographic point of view (where the key sizes are typically greater than 128 bits), it must be noted that the fuzzy vault framework eliminates the key management problem, which is a major issue in practical cryptosystems.

The min-entropy under the uniform assumption also corresponds to the complexity of the brute force attack. Clancy et al. [42] proposed a fingerprint-based fuzzy vault

implementation where the complexity of brute force attack was estimated to be 69 bits. The complexity of brute force attack in our implementation is significantly lower compared to that of Clancy et al. [42] due to the two main reasons. Firstly, recall that we employ CRC-based error detection instead of Reed-Solomon polynomial reconstruction used in [42]. While CRC-based error detection improves the genuine accept rate significantly, only $(n+1)$ genuine points need to be identified for successful decoding. On the other hand, more than $\frac{(r+n)}{2}$ genuine points need to be identified for successfully decoding the vault in [42], which makes it more difficult for an adversary to decode the vault by a brute force attack. Secondly, in the implementation proposed by Clancy et al. [42], chaff points are continuously added until it becomes impossible to add any more chaff points without violating the minimum distance constraint. In our implementation, we restrict the number of chaff points to approximately 10 times the number of genuine points. While adding more chaff points may increase the security of the system, it also increases the memory required to store the vault. Moreover, Chang et al. [24] show that as the number of chaff points is increased, the amount of free area available for adding new chaff points decreases because of the minimum distance constraint. As a result, it may be easier for an adversary to identify some of the chaff points in the vault [24], thereby limiting the security.

Given a database of fingerprints, one can also compute the average min-entropy of the proposed vault implementation as follows. We estimate the distribution of minutiae location and minutiae orientation using the mixture models proposed by Zhu et al. [215]. Based on these estimated distributions, we can compute the min-entropy for each vault and thereby the average min-entropy for that database using

equations (5.10) and (5.7), respectively. For the MSU-DBI database, the average min-entropy when $r$ is between 24 and 36, $n = 10$ and $t = 336$ is approximately 27 bits. This large entropy loss (from $\approx$ 40 bits in the brute force case to $\approx$ 27 bits) is mainly because in our implementation, we assume that the spatial distribution of minutiae in a fingerprint image is uniform and use this property in the generation of chaff points. Due to this assumption, the chaff points in our implementation do not follow the true minutiae distribution, which was shown by Zhu et al. [215] to follow a mixture model. For instance, the minutiae tend to mostly fall around the center of the fingerprint image. However, the chaff points can fall anywhere in the fingerprint image including regions close to the image boundaries (see Figure 5.12(c)). Thus, it is easier to separate the chaff points from the genuine points in our implementation. One way to improve the security of our vault implementation is to estimate the statistical distribution of minutiae during vault encoding and use the estimated minutiae distribution for the generation of chaff points.

Our automatic fingerprint vault implementation is based on the assumption that high curvature points do not reveal any information about the minutiae and it is not possible to estimate the orientation field using only the high curvature points. However, suppose a smart attacker is able to extract the orientation field from the high curvature points and uses it to identify the chaff points. We can still defend against such an attack by introducing some additional chaff points that are consistent with the orientation field (i.e., the location of such a chaff point is random, but its direction is determined by the orientation field) to the set of completely random chaff points.

### 5.7.2 Iris Cryptosystem

In the proposed iris cryptosystem, the helper data consists of two components, namely, the transformed iriscode template $I_*$ and the vault $V$ that secures the transformation key $K_1$ used to obtain $I_*$. Since the transformation key $K_1$ is independent of the template iriscode, it can be generated from a uniform distribution. Therefore, the min-entropy of $K_1$ given $V$ ($H_\infty(K_1|V)$) can be computed using equation (5.11). In our implementation of the vault for the iris modality, $r = 48$, $n = 10$ and $t = 548$. Hence, $H_\infty(K_1|V)$ is approximately 40 bits.

Since we use a single exclusive-OR operation to obtain $I_*$, the min-entropy of template iriscode $I_T$ given $I_*$ ($H_\infty(I_T|I_*)$) depends only on the redundancy added to the key $K_1$ by the BCH encoder. Hao et al. [76] have estimated that in the worst case of an adversary having perfect knowledge of the correlation between the iriscode bits, the inherent uncertainty in a iriscode template is approximately 249 bits. They also showed that if a coding scheme can correct up to $w$ bits in the iriscode template, the entropy of the iriscode template given the transformed template is approximately $\log\left(2^{249}/\binom{249}{w}\right)$ bits. In our implementation, the BCH coding scheme can correct up to 25% of the errors, which corresponds to approximately $w = 62$ bits (out of 249). Therefore, entropy of the $I_T$ given $I_*$ ($H_\infty(I_T|I_*)$) is approximately 52 bits. The overall security of the iris cryptosystem is given by $min\left(H_\infty(K_1|V), H_\infty(I_T|I_*)\right) \approx min(40, 52) \approx 40$ bits.

It must be emphasized that while the inherent entropy of an iriscode template is approximately 249 bits, a system that stores the iriscode template in plaintext form

is secure only when the adversary does not know the template. Once the template is gleaned by the adversary, the system effectively offers no security. However, even when the helper data extracted from the iriscode template is known to the adversary, the proposed iris cryptosystem provides a security of approximately 40 bits. The security of a comparable key-binding cryptosystem proposed by Hao et al. [76] is approximately 44 bits.

### 5.7.3 Multimodal Vault

In the case of the multimodal vault, both the minutiae template $M^T$ and the transformation key $K_1$ used in the iris cryptosystem are secured using a single vault $V$. Therefore, decoding the vault reveals both the $M^T$ and $K_1$ (and consequently $I_T$). Hence, the overall security of the system is given by $min\left(H_\infty(M^T, K_1|V), H_\infty(I_T|I_*)\right)$. In the multimodal vault, $t = 884$, $n = 13$, and the number of genuine points, $r$, is 84 (36 from the fingerprint modality and 48 from the iris modality). If we assume that the minutiae are uniformly distributed, $H_\infty(M^T, K_1|V)$ is approximately 49 bits. Hence, the overall security of the multimodal vault is approximately $min(49, 52) = 49$ bits.

On the other hand, suppose that we construct two separate vaults $V_F$ and $V_I$ for the fingerprint and iris modalities, respectively. In this scenario, the overall security of the system is given by $min\left(\log\left(2^{H_\infty(M^T,|V_F)} + 2^{H_\infty(K^1,|V_I)}\right), H_\infty(I_T|I_*)\right)$, which is approximately 41 bits for the same number of chaff points (300 for fingerprint and 500 for iris). Thus, the multimodal vault provides a significantly higher security compared to storing the individual templates using separate vaults.

Table 5.7: Security of the proposed fuzzy vault implementations. Here, the security is measured in terms of $H_\infty(T|V)$, which represents the average min-entropy of the biometric template $T$ given the vault $V$. The parameters $t$, $r$ and $n$ represent the total number of points in the vault (genuine and chaff), number of genuine points in the vault and the degree of the polynomial used in the vault, respectively.

| Modality | Assumptions | Parameters | | | Security (bits) |
|---|---|---|---|---|---|
| | | $t$ | $r$ | $n$ | |
| Fingerprint | Uniform distribution of minutiae | 330 | 30 | 10 | 40 |
| | Distribution of minutiae follows mixture model [215] | 336 | 24-26 | 10 | 27 |
| Iris | Iriscode has inherent entropy of 249 bits [76]; BCH code corrects up to 25% of the errors | 548 | 48 | 10 | 40 |
| Fingerprint + Iris | Uniform distribution of minutiae; iriscode has inherent entropy of 249 bits [76]; BCH code corrects up to 25% of the errors | 884 | 84 | 13 | 49 |

The security of the proposed vault implementations is summarized in Table 5.7. Apart from the attacks that depend on separating the genuine and chaff points in the vault, there are other specific attacks that can be staged against a fuzzy vault, e.g., *attacks via record multiplicity*, *stolen key inversion attack* and *blended substitution attack* [175]. If an adversary has access to two different vaults (say from two different applications) obtained from the same biometric data, he can easily identify the genuine points in the two vaults and decode the vault [106]. Thus, the fuzzy vault scheme does not provide revocability. An advantage of the fuzzy vault (key binding) scheme is that instead of providing a "Match/Non-match" decision, the vault decoding outputs a key that is embedded in the vault. This key can be used in a variety of ways to authenticate a person (e.g., digital signature, document encryption/decryption etc.). In a stolen key inversion attack, if an adversary somehow recovers the key embedded in the vault, he can decode the vault to obtain the biometric template. Since the vault contains a large number of chaff points, it is possible for an adversary to substitute a few points in the vault using his own biometric features. This allows both the genuine user and the adversary to be successfully authenticated using the same identity, and such an attack is known as blended substitution. To counter these attacks, Nandakumar et al. [145] proposed a hybrid approach where (i) biometric features are first "salted" based on a user password, (ii) the vault is constructed using the salted template and (iii) the vault is encrypted using a key derived from the password. While salting prevents attacks via record multiplicity and provides revo-

cability, encryption provides resistance against blended substitution and stolen key inversion attacks. Moreover, the distribution of biometric features after salting can be expected to be more similar to the uniform distribution than the original feature distribution, which improves the security of the vault.

## 5.8 Summary

Biometric systems are being widely used to achieve reliable user authentication and these systems will proliferate into the core information infrastructure of the (near) future. When this happens, it is crucial to ensure that biometric authentication will be secure. Fuzzy vault is one of the most comprehensive mechanisms for secure biometric authentication. We have implemented a *fully automatic* and *practical multibiometric* fuzzy vault system that can easily secure multiple biometric templates of a user such as fingerprint minutiae and iriscodes as a single entity. The main challenge in the implementation of a fingerprint-based fuzzy vault is the alignment of the query with the transformed template stored in the vault. We use high curvature points derived from the orientation field to align the template and query minutiae sets without leaking any information about the minutiae. We have also developed an iris cryptosystem that uses both salting and fuzzy vault frameworks to secure the iriscode template. Finally, we have demonstrated that templates from multiple biometric sources such as two impressions from the same finger, left and right index fingers and different modalities like fingerprint and iris can be secured using a multibiometric vault. Our experimental evaluation indicates that a multibiometric vault provides both higher

genuine accept rate and higher security.

# Chapter 6

# Conclusions and Future Research

## 6.1 Conclusions

The design of a multibiometric system is a challenging task due to heterogeneity of the biometric sources in terms of their type of information, the magnitude of information content, correlation among the different sources and conflicting performance requirements of the practical applications. In this thesis, we have developed a comprehensive statistical framework for score fusion in multibiometric systems and a framework for multibiometric template security.

First, we developed a principled approach for score level fusion in a multibiometric verification system that employs the likelihood ratio test. The likelihood ratio based approach provides optimal fusion performance when the match score densities are estimated accurately. We investigated two different techniques for density estimation, namely, a non-parametric approach based on kernel density estimation (KDE) and a semi-parametric approach based on finite Gaussian mixture models

(GMM). Both these techniques are quite effective in modeling the genuine and impostor score densities and achieve consistently high recognition rates across three different multibiometric databases without the need for any parameter tuning. But, we believe that the GMM-based approach is simpler to implement than KDE. We also observed that modeling the correlation between the matchers did not lead to any significant improvement in the recognition performance. Therefore, assuming independence between matchers and estimating the joint density as a product of the marginal densities may be appropriate in scenarios where the individual matchers are quite accurate (less than 5% equal error rate) and the difference between genuine and impostor correlations is low.

Further, we have demonstrated that the likelihood ratio based fusion scheme can easily take into account ancillary information such as biometric image quality to improve the recognition performance. Pairwise quality indices that estimate the quality of the template and the query images as a single value were developed for the fingerprint and iris modalities. We have also shown that the marginal likelihood ratios of the individual matchers can be used as inputs to a binary decision tree classifier to design a sequential multibiometric system.

When the match scores of individual users are assumed to be independent and identically distributed, the genuine and impostor densities estimated in the verification scenario can also be used for likelihood ratio based fusion in the multibiometric identification scenario. Moreover, we have shown that the likelihood ratios computed based on the match scores can be combined with the rank-based posterior probabilities and the hybrid rank and score level fusion scheme achieves high recognition

performance in multibiometric identification systems.

To address the problem of template security in a multibiometric system, we have developed a framework for securing multiple biometric templates of a user in the multibiometric system as a single entity. This is achieved by generating a single multibiometric template from different biometric sources using feature level fusion and securing the multibiometric template using the fuzzy vault construct. We have also implemented the fuzzy vault system for securing the fingerprint minutiae and iriscode templates individually. The problem of alignment in the fingerprint-based fuzzy vault is handled by storing high curvature points extracted from the orientation field as additional helper data. A salting transformation based on a transformation key was used to indirectly convert the fixed-length binary vector representation of iriscode into an unordered set representation that can be secured using the fuzzy vault. Finally, we have shown that the multibiometric vault can secure templates from different biometric sources such as multiple fingerprint impressions, multiple fingers and multiple modalities such as fingerprint and iris. We have also demonstrated that the multibiometric vault provides better recognition performance and security compared to the individual vaults.

## 6.2 Future Research Directions

While we have made significant progress in the development of fusion strategies and template protection schemes that facilitate the design of reliable and secure multibiometric systems, we believe that the techniques proposed in this thesis can be further

expanded and refined in the following ways.

- The fusion strategies proposed in this thesis can be considered as global techniques in the sense that no user-specific information is used in developing these schemes. This is implicitly based on the assumption that the discriminatory information provided by the individual biometric sources is identical across all users. However, it is well-known that there are inherent differences in the "recognizability" of the different users [54] and user-specific fusion techniques can further improve the recognition performance of the multibiometric system [63, 96, 190]. User-specific fusion can be achieved within the likelihood ratio framework by learning user-specific match score densities when sufficient training data is available for each user.

- Our experimental results indicate that modeling the correlation between the match scores of the different matchers does not result in any significant improvement in the fusion performance. A theoretical model that establishes the effect of match score correlation on fusion performance is needed to validate this observation.

- The density estimation techniques used in this thesis operate in the batch mode where the genuine and impostor score densities are estimated only once during the system design phase based on the complete training data. When there is a significant change in the matcher characteristics, the density estimation process needs to be repeated again starting from scratch with new training data. Moreover, in practical multibiometric systems, additional training data

may become available during the operation of the system. Therefore, it may be beneficial to use incremental GMM learning algorithms [216] or Bayesian adaptation [163] schemes that can update the score densities when additional training data becomes available without the need for re-training.

- Apart from the location and orientation attributes of a minutia point, many minutiae-based fingerprint matchers use additional attributes like minutia type, ridge counts, ridge curvature, ridge density and local texture features [61] to achieve high recognition rates. These attributes could also be incorporated into the fingerprint-based fuzzy vault framework. Addition of new attributes will not only increase the number of possible chaff points that can be added to the vault but also decrease the decoding complexity for genuine users and reduce the false accept rate. The integration of other common biometric modalities such as face and voice in the multibiometric vault framework also requires further investigation.

- A well-known limitation of the fuzzy vault framework is its dependence on chaff points to achieve security. Therefore, other biometric cryptosystems that do not involve chaff points could be considered for securing the biometric templates.

- Finally, a formal model for cost-benefit analysis of a multibiometric system based on parameters such as performance gain (reduction in FRR/FAR), throughput, physical cost of the system and security needs to be developed in order to enable biometric system developers to rapidly design a multibiometric system that is most appropriate for the application on hand.

# Appendices

# A Databases

## A.1 Multibiometric Databases

We use two public-domain match score databases, namely, NIST-BSSR1 and XM2VTS-Benchmark databases to benchmark the various fusion strategies considered in this thesis. The performance of the quality-based product fusion rule was evaluated only on the WVU-Multimodal database since the other databases do not contain raw fingerprint and iris images to enable us to estimate the biometric sample quality. The performance of the proposed fusion rules was also evaluated on the in-house MSU-Multimodal database and the results of the evaluation on this database has been reported in [48]. Table 1 presents a summary of the multibiometric databases used in this thesis.

### NIST-BSSR1

The NIST Biometric Scores Set - Release I (NIST-BSSR1) [151] has three partitions. The first partition is the NIST-Multimodal database, which consists of 517 users with two fingerprint and two face scores. One fingerprint score was obtained by comparing a pair of impressions of the left index finger, and the second score was obtained by comparing impressions of the right index finger. Two different face matchers were applied to compute the similarity between frontal face images. The NIST-Multimodal database is a "true" multimodal database in the sense that the fingerprint and face images used for computing the genuine match scores were derived from the same individual. The second partition of NIST-BSSR1 is the NIST-Fingerprint database,

which is an example of multi-instance(finger) biometric system. This partition consists of scores from left and right index fingerprint matches of $6,000$ individuals. The third partition is the NIST-Face database, which consists of scores from two face matchers applied on three frontal face images from $3,000$ individuals.

**XM2VTS-Benchmark**

The XM2VTS-Benchmark database [154] consists of five face matchers and three speech matchers and was partitioned into training, fusion development and fusion evaluation sets according to the Lausanne Protocol-1 (see [154] for details).

**WVU-Multimodal**

The West Virginia University multimodal database (WVU-Multimodal) consists of 320 virtual subjects (subjects created by randomly pairing a user from one unimodal database (e.g., iris) with a user from another database (e.g., fingerprint)) with five samples each of fingerprint and iris modalities. Minutiae-based fingerprint matcher [93] and Iriscode [50] based iris matcher were used for computing the match scores.

**MSU-Multimodal**

The MSU-Multimodal database [90] consists of 100 virtual subjects, each providing five samples of face, fingerprint (left-index) and hand-geometry modalities. Face images were represented as eigenfaces [193] and the Euclidean distance between the eigen-coefficients of the template-query pair was used as the distance metric. Minutia points were extracted from fingerprint images and the elastic string matching tech-

nique [86] was used for computing the similarity between two minutia point patterns. Fourteen features describing the geometry of the hand shape [98] were extracted from the hand images and Euclidean distance was computed for each template-query pair.

## A.2 Fingerprint Databases

The performance of the fingerprint-based fuzzy vault implementation has been evaluated on FVC2002-DB2 [128] and MSU-DBI [94] fingerprint databases, which are summarized in Table 2.

### MSU-DBI

The MSU-DBI database contains two pairs of impressions for each of the 160 users and these two pairs were collected six weeks apart. Further, images from four different fingers (two index and two middle fingers) are available for each user. Hence, this database is suitable to study the multiple finger and multiple impression scenarios in the fuzzy vault implementation. We use only the impressions from right and left index fingers in our experiments.

### FVC2002-DB2

FVC2002-DB2 was one of the benchmark databases used in the Fingerprint Verification Competition 2002 [128]. The FVC2002-DB2 consists of 100 fingers with 8 impressions per finger obtained using an optical fingerprint sensor. This database was selected because it is a public-domain database and the images are of relatively good quality. Among the 8 impressions available for each finger in FVC2002-DB2,

Table 1: Summary of multibiometric databases. Note that the NIST-Multimodal, NIST-Fingerprint and NIST-Face databases are different partitions of the NIST Biometric Score Set Release-1.

| **Database** | **Biometric Traits** | **No. of matchers** | **No. of users** |
|---|---|---|---|
| NIST-Multimodal | Fingerprint (Two fingers) Face (Two matchers) | 4 | 517 |
| NIST-Fingerprint | Fingerprint (Two fingers) | 2 | 6,000 |
| NIST-Face | Face (Two matchers) | 2 | 3,000 |
| XM2VTS-Benchmark | Face (Five matchers) Speech (Three matchers) | 8 | 295 |
| WVU-Multimodal | Fingerprint, Iris | 2 | 320 |
| MSU-Multimodal | Fingerprint, Face and Hand-geometry | 3 | 100 |

we use only four impressions (impressions 1, 2, 7 and 8) in our experiments due to the following reason. It is quite reasonable to assume that users in a biometric cryptosystem are co-operative and they are willing to provide good quality biometric data in order to retrieve their cryptographic keys. Impressions 3, 4, 5 and 6 in FVC2002 databases were obtained by requesting the users to provide fingerprints with exaggerated displacement and rotation. Hence, these impressions are not representative for the application under consideration. This explains our choice of impressions 1, 2, 7 and 8.

## A.3 CASIA Iris Database

The performance of the iris cryptosystem has been evaluated on the CASIA iris image database ver 1.0 [39, 126]. This database consists of images from 108 different eyes

Table 2: Summary of fingerprint databases used in the evaluation of fuzzy vault.

| | **FVC2002-DB2** | **MSU-DBI** |
|---|---|---|
| No. of users | 100 | 160 (4 fingers per user) |
| No. of impressions/finger | 8 | 4 |
| Sensor | Biometrika FX2000 (Optical) | Digital Biometrics, Inc. (Optical) |
| Image size | $560 \times 296$ at 569 dpi resolution | $640 \times 480$ at 500 dpi resolution |
| Image quality | Good | Medium |

with 7 images per eye. These 7 samples were collected over two different sessions with 4 samples in one session and 3 in the other. We use one image from each session to evaluate the iris cryptosystem. Recently, Phillips et al. [152] pointed out that the pupil regions in the iris images of this database have been manually edited, which makes it easier to segment the iris region. Hence, they discouraged the use of this database in iris recognition studies. However, we still use the CASIA v1 database in our experiments because our goal is not to develop reliable segmentation or feature extraction algorithms. Rather, the main focus of our work is to develop techniques for securing the given iriscode templates in best possible manner.

# B Algorithms

## B.1 Determining Discrete Components in a Score Distribution

Inputs: $\mathcal{S}$ - Set of match scores, $\alpha$ - Level of significance of chi-squared test, $B$ - Number of bins, $M$ - Number of folds for cross-validation.

Output: $T$ - Threshold to determine discrete components.

1. Initialize $T \leftarrow 1$.

2. Determine the collection $\mathcal{C}$ of continuous components as follows:

$$\mathcal{C} \equiv \{\, s_0 \,:\, \frac{N(s_0)}{N} < T\}, \tag{1}$$

where $N(s_0)$ is the number of observations in $\mathcal{S}$ that equals $s_0$ and $N$ is the total number of observations in $\mathcal{S}$. The set $\mathcal{C}$ is further divided into $M$ equal and non-overlapping subsets among which one subset is labeled as $\mathcal{C}_V$ while the remaining $M-1$ subsets are combined to form the set $\mathcal{C}_T$. The set $\mathcal{C}_T$ is used for density estimation and $\mathcal{C}_V$ is used for validating the estimated density.

3. Based on the data in set $\mathcal{C}_T$, obtain the kernel density estimate of $f_C(s)$, $\hat{f}_C(s)$, using equation (3.4). The corresponding distribution function $\hat{F}_C(s)$ is obtained as follows.

$$\hat{F}_C(s) = \int_{-\infty}^{s} \hat{f}_C(u)du. \tag{2}$$

4. Use the chi-squared goodness-of-fit test to test the following hypothesis. The null hypothesis is $H_0$*:* $\hat{F}_C(s)$ *is the true distribution of data in* $\mathcal{C}_V$ and the alternative hypothesis is $H_a$*:* $\hat{F}_C(s)$ *is not the true distribution.* The test statistic is given by

$$\chi^2 = \sum_{b=1}^{B} \frac{(O_b - E_b)^2}{E_b}, \tag{3}$$

where $O_b$ is the observed frequency for bin $b$ and $E_b$ is the expected frequency for bin $b$. The $b^{th}$ bin edge is chosen to be $\left[\hat{F}_C^{-1}\left(\frac{b-1}{B}\right), \hat{F}_C^{-1}\left(\frac{b}{B}\right)\right)$. Due to this particular choice of bin edges, it follows that $E_b = N_V/B$, where $N_V$ is the number of observations in $\mathcal{C}_V$.

5. Repeat steps 3 and 4 $M$ times; each time a different subset of $\mathcal{C}$ is chosen as $\mathcal{C}_V$ while the remaining subsets form $\mathcal{C}_T$. The average test statistic $\chi^2_{avg}$ is computed.

6. Let $\chi^2_{(\alpha,B-d-1)}$ be the value such that a fraction $\alpha$ of the area under the $\chi^2$ distribution with $B-d-1$ degrees of freedom lies to the right of $\chi^2_{(\alpha,B-d-1)}$, where $d$ is the number of estimated parameters. Since we estimate only the bandwidth of the kernel from the data, we set the value of $d$ to 1. If $\chi^2_{avg} > \chi^2_{(\alpha,B-d-1)}$, we reject the null hypothesis, set $T \leftarrow argmax_{s_0 \in \mathcal{C}} N(s_0)/N$ and return to step 2. Else, we output the value of $T$.

## B.2 Juels-Sudan Vault Encoding

**Public Parameters:** A field $\mathcal{F}$
**Input:** Parameters $n, r$ and $s$ such that $0 < n < r \ll s$; a secret $K$; a set $X = \{x_i\}_{i=1}^{r}$ representing the user's biometric template such that $x_i \in \mathcal{F}$ and $x_i \neq x_j \ \forall \ i \neq j, \ i, j = 1, 2, \cdots, r$
**Output:** A vault $V = \left\{\left(a_j, b_j\right)\right\}_{j=1}^{t}$, where $t = r + s$

$\mathcal{P} \leftarrow$ ENCODESECRET$(K)$
$L, C, Y \leftarrow \phi$
**for** $j = 1$ to $r$ **do**
  $\left(a_j, b_j\right) \leftarrow \left(x_j, \mathcal{P}(x_j)\right)$
  $L \leftarrow L \cup (a_j, b_j)$
**end for**
**for** $j = r + 1$ to $t$ **do**
  $y_j \in \mathcal{F} - (X \cup Y)$
  $Y \leftarrow Y \cup y_j$
  $z_j \in \mathcal{F} - \{\mathcal{P}(y_j)\}$
  $\left(a_j, b_j\right) \leftarrow \left(y_j, z_j\right)$
  $C \leftarrow C \cup (a_j, b_j)$
**end for**
$V^{'} \leftarrow L \cup C$
$V \leftarrow$ PERMUTE$(V^{'})$
Return $V$

## B.3 Juels-Sudan Vault Decoding

**Public Parameters:** A field $\mathcal{F}$
**Input:** Parameters $n, r$ and $s$ such that $0 < n < r \ll s$; a set $X^{'} = \{x_i^{'}\}_{i=1}^{r}$ representing the user's biometric query such that $x_i^{'} \in \mathcal{F}$ and $x_i^{'} \neq x_j^{'}\ \forall\ i \neq j,\ i, j = 1, 2, \cdots, r$; a vault $V = \left\{\left(a_j, b_j\right)\right\}_{j=1}^{t}$
**Output:** A secret $K$ or *null*

$L^{'} \leftarrow \phi$
**for** $i = 1$ to $r$ **do**
  $\left(a_i^{'}, b_i^{'}\right) \leftarrow null$
  **for** $j = 1$ to $t$ **do**
    **if** $(x_i^{'} = a_j)$ **then**
      $\left(a_i^{'}, b_i^{'}\right) \leftarrow \left(a_j, b_j\right)$
      break
    **end if**
  **end for**
  $L^{'} \leftarrow L^{'} \cup (a_i^{'}, b_i^{'})$
**end for**
$\mathcal{P} \leftarrow$ RSDECODE($L^{'}$)
**if** $(\mathcal{P} = null)$ **then**
  Return *null*
**else**
  $K \leftarrow$ DECODESECRET($\mathcal{P}$)
  Return $K$
**end if**

## B.4 Alignment using ICP

**Input:** Parameters $k_{max}$ and $D_{stop}$ ; Template helper data $H^T = \{h_i^T\}_{i=1}^{R^T}$;
Query helper data set $H^Q = \{h_j^Q\}_{j=1}^{R^Q}$

**Output:** A transformation $F$ that best aligns $H^Q$ with $H^T$

$k \leftarrow 0$
$H_0^Q \leftarrow H^Q$
$MWD_{old} \leftarrow 10^6$
**while** $(k < k_{max})$ **do**
  $k \leftarrow k + 1$
  **if** $(k = 1)$ **then**
    $F^{'} \leftarrow$INITTRANS$\left(H^T, H^Q\right)$
  **else**
    $F^{'} \leftarrow$TRANS$\left(H^{T|Q}, H^Q\right)$
  **end if**
  $H^Q \leftarrow F^{'}\left(H^Q\right)$
  $H^{T|Q} \leftarrow \phi$
  **for** $j = 1$ to $R^Q$ **do**
    $i = \text{argmin}_{i'}\, D_H\left(h_{i'}^T, h_j^Q\right)$
    $h_j^{T|Q} = h_i^T$
    $H^{T|Q} \leftarrow H^{T|Q} \cup h_j^{T|Q}$
  **end for**
  $MWD_{new} \leftarrow \frac{1}{R^Q}\sum_{j=1}^{R^Q} D_H\left(h_j^{T|Q}, h_j^Q\right)$
  **if** $\left((MWD_{old} - MWD_{new}) < D_{stop}\right)$ **then**
    break
  **else**
    $MWD_{old} \leftarrow MWD_{new}$
  **end if**
**end while**
$F \leftarrow$TRANS$\left(H^{T|Q}, H_0^Q\right)$
Return $F$

# Bibliography

# Bibliography


[1] Advanced Encryption Standard, November 2001.

[2] A. Adler. Sample images can be independently restored from face recognition templates. In *Proceedings of Canadian Conference on Electrical and Computer Engineering*, volume 2, pages 1163–1166, Montreal, Canada, May 2003.

[3] Andy Adler. Images can be Regenerated from Quantized Biometric Match Score Data. In *Proceedings Canadian Conference on Electrical and Computer Engineering*, pages 469–472, Niagara Falls, Canada, May 2004.

[4] A. Arakala, J. Jeffers, and K. J. Horadam. Fuzzy Extractors for Minutiae-Based Fingerprint Authentication. In *Proceedings of Second International Conference on Biometrics*, pages 760–769, Seoul, South Korea, August 2007.

[5] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces versus Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(7):711–720, 1997.

[6] S. Ben-Yacoub, Y. Abdeljaoued, and E. Mayoraz. Fusion of Face and Speech Data for Person Identity Verification. *IEEE Transactions on Neural Networks*, 10(5):1065–1075, September 1999.

[7] E. R. Berlekamp. *Algebraic Coding Theory.* McGraw Hill, 1968.

[8] P. Besl and N. McKay. A Method for Registration of 3-D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, February 1992.

[9] B. Bhanu and X. Tan. Fingerprint Indexing Based on Novel Features of Minutiae Triplets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):616–622, May 2003.

[10] E. S. Bigun, J. Bigun, B. Duc, and S. Fischer. Expert Conciliation for Multimodal Person Authentication Systems using Bayesian Statistics. In *Proceedings of First International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pages 291–300, Crans-Montana, Switzerland, March 1997.

[11] D. Bleichenbacher and P. Q. Nguyen. Noisy Polynomial Interpolation and Noisy Chinese Remaindering. In *Proceedings of Nineteenth IACR Eurocrypt*, pages 53–69, Bruges, Belgium, May 2000.

[12] R. Bolle, J. Connell, S. Pankanti, N. Ratha, and A. Senior. The Relationship Between the ROC Curve and the CMC. In *Proceedings of Fourth IEEE Workshop on Automatic Identification Advanced Technologies (AutoID)*, pages 15–20, Buffalo, USA, October 2005.

[13] T. E. Boult, W. J. Scheirer, and R. Woodworth. Fingerprint Revocable Biotokens: Accuracy and Security Analysis. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2007.

[14] X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, and A. Smith. Secure Remote Authentication Using Biometric Data. In *Advances in Cryptology–EUROCRYPT 2005*, pages 147–163, Aarhus, Denmark, May 2005.

[15] R. Brunelli and D. Falavigna. Person Identification Using Multiple Cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(10):955–966, October 1995.

[16] I. R. Buhan, J. M. Doumen, P. H. Hartel, and R. N. J. Veldhuis. Fuzzy Extractors for Continuous Distributions. In *Proceedings of ACM Symposium on Information, Computer and Communications Security*, pages 353–355, Singapore, March 2007.

[17] I. R. Buhan, J. M. Doumen, P. H. Hartel, and R. N. J. Veldhuis. Secure Ad-hoc Pairing with Biometrics: SAfE. In *Proceedings of First International Workshop on Security for Spontaneous Interaction*, pages 450–456, Innsbruck, Austria, September 2007.

[18] W. E. Burr, D. F. Dodson, and W. T. Polk. Information Security: Electronic Authentication Guideline. Technical Report Special Report 800-63, NIST, April 2006.

[19] E. Camlikaya, A. Kholmatov, and B. Yanikoglu. Multimodal Biometric Templates Using Fingerprint and Voice. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification V (To appear)*, Orlando, USA, March 2008.

[20] W. M. Campbell, D. A. Reynolds, and J. P. Campbell. Fusing Discriminative and Generative Methods for Speaker Recognition: Experiments on Switchboard and NFI/TNO Field Data. In *Odyssey: The Speaker and Language Recognition Workshop*, pages 41–44, Toledo, Spain, May 2004.

[21] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni. Fingerprint Image Reconstruction From Standard Templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9):1489–1503, 2007.

[22] A. Cavoukian and A. Stoianov. Biometric Encryption: A Positive-Sum Technology that Achieves Strong Authentication, Security and Privacy. Technical report, Office of the Information and Privacy Commissioner of Ontario, March 2007.

[23] E. C. Chang and S. Roy. Robust Extraction of Secret Bits From Minutiae. In *Proceedings of Second International Conference on Biometrics*, pages 750–759, Seoul, South Korea, August 2007.

[24] E.-C. Chang, R. Shen, and F. W. Teo. Finding the Original Point Set Hidden Among Chaff. In *Proceedings of ACM Symposium on Information, Computer and Communications Security*, pages 182–188, Taipei, Taiwan, 2006.

[25] K. Chang, K. W. Bowyer, S. Sarkar, and B. Victor. Comparison and Combination of Ear and Face Images in Appearance-based Biometrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(9):1160–1165, September 2003.

[26] K. I. Chang, K. W. Bowyer, and P. J. Flynn. An Evaluation of Multimodal 2D+3D Face Biometrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4):619–624, April 2005.

[27] K. I. Chang, K. W. Bowyer, P. J. Flynn, and X. Chen. Multibiometrics Using Facial Appearance, Shape and Temperature. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, pages 43–48, Seoul, Korea, May 2004.

[28] Y.-J. Chang, W. Zhang, and T. Chen. Biometrics Based Cryptographic Key Generation. In *Proceedings of IEEE Conference on Multimedia and Expo*, volume 3, pages 2203–2206, Taipei, Taiwan, June 2004.

[29] O. Chapelle and V. Vapnik. Model Selection for Support Vector Machines. In *Advances in Neural Information Processing Systems 12 (NIPS)*, pages 230–236, Colorado, USA, November-December 1999.

[30] K. Chen, L. Wang, and H. Chi. Methods of Combining Multiple Classifiers with Different Features and their Applications to Text-Independent Speaker Identification. *International Journal of Pattern Recognition and Artificial Intelligence*, 11(3):417–445, 1997.

[31] X. Chen, P. J. Flynn, and K. W. Bowyer. IR and Visible Light Face Recognition. *Computer Vision and Image Understanding*, 99(3):332–358, September 2005.

[32] Y. Chen, S. C. Dass, and A. K. Jain. Fingerprint Quality Indices for Predicting Authentication Performance. In *Proceedings of Fifth International Conference on Audio- and Video-Based Biometric Person Authentication*, pages 160–170, Rye Brook, USA, July 2005.

[33] Y. Chen, S. C. Dass, and A. K. Jain. Localized Iris Image Quality Using 2-D Wavelets. In *IAPR International Conference on Biometrics (ICB)*, pages 373–381, Hong Kong, China, January 2006.

[34] U. Cherubini, E. Luciano, and W. Vecchiato. *Copula Methods in Finance*. Wiley, 2004.

[35] D. Chetverikov, D. Svirko, D. Stepanov, and P. Krsek. The Trimmed Iterative Closest Point Algorithm. In *Proceedings of International Conference on Pattern Recognition*, pages 545–548, Quebec City, Canada, August 2002.

[36] M. Cheung, K. Yiu, M. Mak, and S. Kung. Multi-Sample Fusion with Constrained Feature Transformation for Robust Speaker Verification. In *Eighth International Conference on Spoken Language Processing (ICSLP)*, pages 1813–1816, Jeju Island, Korea, October 2004.

[37] C. C. Chibelushi, J. S. D. Mason, and F. Deravi. Feature-level Data Fusion for Bimodal Person Recognition. In *Proceedings of the Sixth International Conference on Image Processing and Its Applications*, volume 1, pages 399–403, Dublin, Ireland, July 1997.

[38] C. S. Chin, A. B. J Teoh, and D. C. L. Ngo. High Security Iris Verification System Based On Random Secret Integration. *Computer Vision and Image Understanding*, 102(2):169–177, May 2006.

[39] Chinese Academy of Sciences. Specification of CASIA Iris Image Database (ver 1.0). `http://www.nlpr.ia.ac.cn/english/irds/irisdatabase.htm`, March 2007.

[40] K. Choi, H. Choi, and J. Kim. Fingerprint Mosaicking by Rolling and Sliding. In *Proceedings of Fifth International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pages 260–269, Rye Brook, USA, July 2005.

[41] Y. Chung, D. Moon, S. Lee, S. Jung, T. Kim, and D. Ahn. Automatic Alignment of Fingerprint Features for Fuzzy Fingerprint Vault. In *Proceedings of Conference on Information Security and Cryptology*, pages 358–369, Beijing, China, December 2005.

[42] T. Clancy, D. Lin, and N. Kiyavash. Secure Smartcard-Based Fingerprint Authentication. In *Proceedings of ACM SIGMM Workshop on Biometric Methods and Applications*, pages 45–52, Berkley, USA, November 2003.

[43] T. Connie, A. B. J Teoh, M. Goh, and D. C. L. Ngo. PalmHashing: A Novel Approach for Cancelable Biometrics. *Information Processing Letters*, January 2005.

[44] IBM Corporation. The Consideration of Data Security in a Computer Environment. Technical Report G520-2169, IBM, White Plains, USA, 1970.

[45] J. Czyz, J. Kittler, and L. Vandendorpe. Multiple Classifier Combination for Face-based Identity Verification. *Pattern Recognition*, 37(7):1459–1469, July 2004.

[46] S. C. Dass. Markov Random Field Models for Directional Field and Singularity Extraction in Fingerprint Images. *IEEE Transactions on Image Processing*, 13(10):1358–1367, October 2004.

[47] S. C. Dass and A. K. Jain. Fingerprint Classification Using Orientation Field Flow Curves. In *Proceedings of Indian Conference on Computer Vision, Graphics and Image Processing*, pages 650–655, Kolkata, India, December 2004.

[48] S. C. Dass, K. Nandakumar, and A. K. Jain. A Principled Approach to Score Level Fusion in Multimodal Biometric Systems. In *Proceedings of Fifth International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA)*, pages 1049–1058, Rye Brook, USA, July 2005.

[49] J. Daugman. Combining Multiple Biometrics. Available at `http://www.cl.cam.ac.uk/users/jgd1000/combine/combine.html`, 2000.

[50] J. Daugman. How Iris Recognition Works? *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):21–30, 2004.

[51] G. I. Davida, Y. Frankel, and B. J. Matt. On Enabling Secure Applications Through Off-Line Biometric Identification. In *Proceedings of IEEE Symposium on Security and Privacy*, pages 148–157, Oakland, USA, May 1998.

[52] J. De Boer, A. M. Bazen, and S. H. Gerez. Indexing Fingerprint Databases Based on Multiple Features. In *Proceedings of Workshop on Circuits, Systems and Signal Processing (ProRISC 2001)*, pages 300–306, Veldhoven, Netherlands, November 2001.

[53] P. A. Devijver and J. Kittler. *Pattern Recognition: A Statistical Approach*. Prentice Hall, 1982.

[54] G. Doddington, W. Liggett, A. Martin, M. Przybocki, and D. Reynolds. Sheep, Goats, Lambs and Wolves: A Statistical Analysis of Speaker Performance in the NIST 1998 Speaker Recognition Evaluation. In *Proceedings of the Fifth International Conference on Spoken Language Processing (ICSLP)*, Sydney, Australia, November/December 1998.

[55] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith. Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data. Technical Report 235, Cryptology ePrint Archive, February 2006. A preliminary version of this work appeared in EUROCRYPT 2004.

[56] S. C. Draper, A. Khisti, E. Martinian, A. Vetro, and J. S. Yedidia. Using Distributed Source Coding to Secure Fingerprint Biometrics. In *Proceedings*

*of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 2, pages 129–132, Hawaii, USA, April 2007.

[57] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley & Sons, 2001.

[58] A. Eriksson and P. Wretling. How Flexible is the Human Voice? A Case Study of Mimicry. In *Proceedings of the European Conference on Speech Technology*, pages 1043–1046, Rhodes, Greece, September 1997.

[59] Y. Fang, T. Tan, and Y. Wang. Fusion of Global and Local Features for Face Verification. In *Sixteenth International Conference on Pattern Recognition (ICPR)*, volume 2, pages 382–385, Quebec City, Canada, August 2002.

[60] G. Feng, K. Dong, D. Hu, and D. Zhang. When Faces are Combined with Palmprints: A Novel Biometric Fusion Strategy. In *First International Conference on Biometric Authentication (ICBA)*, pages 701–707, Hong Kong, China, July 2004.

[61] J. Feng. Combining Minutiae Descriptors for Fingerprint Matching. *Pattern Recognition*, 41(1):342–352, January 2008.

[62] Y. C. Feng and P. C. Yuen. Protecting Face Biometric Data on Smartcard with Reed-Solomon Code. In *Proceedings of CVPR Workshop on Privacy Research In Vision*, page 29, New York, USA, June 2006.

[63] J. Fierrez-Aguilar, D. Garcia-Romero, J. Ortega-Garcia, and J. Gonzalez-Rodriguez. Bayesian Adaptation for User-Dependent Multimodal Biometric Authentication. *Pattern Recognition*, 38(8):1317–1319, August 2005.

[64] J. Fierrez-Aguilar, L. Nanni, J. Lopez-Penalba, J. Ortega-Garcia, and D. Maltoni. An On-line Signature Verification System based on Fusion of Local and Global Information. In *Fifth International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA)*, pages 523–532, Rye Brook, USA, July 2005.

[65] J. Fierrez-Aguilar, J. Ortega-Garcia, J. Gonzalez-Rodriguez, and J. Bigun. Discriminative Multimodal Biometric Authentication based on Quality Measures. *Pattern Recognition*, 38(5):777–779, May 2005.

[66] J. Fierrez-Aguilar, J. Ortega-Garcia, J. Gonzalez-Rodriguez, and J. Bigun. Discriminative Multimodal Biometric Authentication based on Quality Measures. *Pattern Recognition*, 38(5):777–779, May 2005.

[67] M. Figueiredo and A. K. Jain. Unsupervised Learning of Finite Mixture Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):381–396, March 2002.

[68] M. Freire-Santos, J. Fierrez-Aguilar, and J. Ortega-Garcia. Cryptographic Key Generation Using Handwritten Signature. In *Proceedings of Biometric Technologies for Human Identification, Part of SPIE Defense and Security Symposium*, volume 6202, pages 225–231, Orlando, USA, April 2006.

[69] R. Frischholz and U. Dieckmann. BioID: A Multimodal Biometric Identification System. *IEEE Computer*, 33(2):64–68, February 2000.

[70] H. Frohlich and A. Zell. Efficient Parameter Selection for Support Vector Machines in Classification and Regression via Model-based Global Optimization. In *Proceedings of IEEE International Joint Conference on Neural Networks (IJCNN)*, volume 3, pages 1431–1436, Montreal, Canada, July-August 2005.

[71] M. Fuentes, S. Garcia-Salicetti, and B. Dorizzi. On-line Signature Verification: Fusion of a Hidden Markov Model and a Neural Network via a Support Vector Machine. In *Eighth International Workshop on Frontiers in Handwriting Recognition*, pages 253–258, Ontario, Canada, August 2002.

[72] M. D. Garris, C. I. Watson, and C. L. Wilson. Matching Performance for the US-Visit IDENT System Using Flat Fingerprints. Technical Report 7110, National Institute of Standards and Technology (NIST), July 2004. NIST Internal Report 7110.

[73] R. S. Germain, A. Califano, and S. Colville. Fingerprint Matching Using Transformation Parameter Clustering. *IEEE Computational Science and Engineering*, 4(4):42–49, October 1997.

[74] P. Griffin. Optimal Biometric Fusion for Identity Verification. Technical Report RDNJ-03-0064, Identix Corporate Research Center, 2004.

[75] P. Grother and P. J. Phillips. Models of Large Population Recognition Performance. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 68–75, Washington, DC, USA, June-July 2004.

[76] F. Hao, R. Anderson, and J. Daugman. Combining Crypto with Biometrics Effectively. *IEEE Transactions on Computers*, 55(9):1081–1088, September 2006.

[77] K. Harmel and L. Spadanuta. Disney World scans fingerprint details of park visitors. Available at `http://www.boston.com/news/nation/articles/2006/09/03/disney_world_scans_fingerprint_details_of_park_visitors`, September 2006.

[78] W. R. Harrison. *Suspect Documents, their Scientific Examination*. Nelson-Hall Publishers, 1981.

[79] T. K. Ho, J. J. Hull, and S. N. Srihari. Decision Combination in Multiple Classifier Systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(1):66–75, January 1994.

[80] L. Hong and A. K. Jain. Integrating Faces and Fingerprints for Personal Identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1295–1307, December 1998.

[81] L. Hong, A. K. Jain, and S. Pankanti. Can Multibiometrics Improve Performance? In *Proceedings of IEEE Workshop on Automatic Identification Advanced Technologies (AutoID)*, pages 59–64, New Jersey, USA, October 1999.

[82] Y. S. Huang and C. Y. Suen. Method of Combining Multiple Experts for the Recognition of Unconstrained Handwritten Numerals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(1):90–94, January 1995.

[83] S. S. Iyengar, L. Prasad, and H. Min. *Advances in Distributed Sensor Technology*. Prentice Hall, 1995.

[84] A. K. Jain, R. Bolle, and S. Pankanti, editors. *Biometrics: Personal Identification in Networked Society*. Kluwer Academic Publishers, 1999.

[85] A. K. Jain and B. Chandrasekaran. Dimensionality and Sample Size Considerations in Pattern Recognition Practice. In P.R. Krishnaiah and L. N. Kanal, editors, *Handbook of Statistics*, volume 2, pages 835–855. North-Holland, Amsterdam, 1982.

[86] A. K. Jain, L. Hong, and R. Bolle. On-line Fingerprint Verification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):302–314, April 1997.

[87] A. K. Jain, L. Hong, and R. Bolle. On-line Fingerprint Verification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):302–314, April 1997.

[88] A. K. Jain, L. Hong, and Y. Kulkarni. A Multimodal Biometric System using Fingerprint, Face and Speech. In *Second International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA)*, pages 182–187, Washington D.C., USA, March 1999.

[89] A. K. Jain, K. Nandakumar, X. Lu, and U. Park. Integrating Faces, Fingerprints and Soft Biometric Traits for User Recognition. In *Proceedings of ECCV International Workshop on Biometric Authentication (BioAW)*, volume LNCS 3087, pages 259–269, Prague, Czech Republic, May 2004. Springer.

[90] A. K. Jain, K. Nandakumar, and A. Ross. Score Normalization in Multimodal Biometric Systems. *Pattern Recognition*, 38(12):2270–2285, December 2005.

[91] A. K. Jain and S. Pankanti. A Touch of Money. *IEEE Spectrum*, 3(7):22–27, 2006.

[92] A. K. Jain, S. Pankanti, S. Prabhakar, L. Hong, and A. Ross. Biometrics: A Grand Challenge. In *Proceedings of International Conference on Pattern Recognition (ICPR)*, volume 2, pages 935–942, Cambridge, UK, August 2004.

[93] A. K. Jain, S. Prabhakar, and S. Chen. Combining Multiple Matchers for a High Security Fingerprint Verification System. *Pattern Recognition Letters*, 20(11-13):1371–1379, November 1999.

[94] A. K. Jain, S. Prabhakar, and A. Ross. Fingerprint Matching: Data Acquisition and Performance Evaluation. Technical Report TR99-14, Michigan State University, 1999.

[95] A. K. Jain and A. Ross. Fingerprint Mosaicking. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 4, pages 4064–4067, Orlando, USA, May 2002.

[96] A. K. Jain and A. Ross. Learning User-specific Parameters in a Multibiometric System. In *Proceedings of International Conference on Image Processing (ICIP)*, pages 57–60, Rochester, USA, September 2002.

[97] A. K. Jain and A. Ross. Multibiometric Systems. *Communications of the ACM, Special Issue on Multimodal Interfaces*, 47(1):34–40, January 2004.

[98] A. K. Jain, A. Ross, and S. Pankanti. A Prototype Hand Geometry-based Verification System. In *Proceedings of Second International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA)*, pages 166–171, Washington D.C., USA, March 1999.

[99] A. K. Jain, A. Ross, and S. Prabhakar. An Introduction to Biometric Recognition. *IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Image- and Video-Based Biometrics*, 14(1):4–20, January 2004.

[100] D. S. Jeong, H.-A. Park, K. R. Park, and J. Kim. Iris Recognition in Mobile Phone Based on Adaptive Gabor Filter. In *Proceedings of IAPR International Conference on Biometrics (ICB)*, pages 457–463, Hong Kong, China, January 2006.

[101] X. Jiang and W. Ser. Online fingerprint template improvement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1121–1126, August 2002.

[102] A. Juels and M. Sudan. A Fuzzy Vault Scheme. In *Proceedings of IEEE International Symposium on Information Theory*, page 408, Lausanne, Switzerland, 2002.

[103] A. Juels and M. Wattenberg. A Fuzzy Commitment Scheme. In *Proceedings of Sixth ACM Conference on Computer and Communications Security*, pages 28–36, Singapore, November 1999.

[104] A. Kale, A. K. RoyChowdhury, and R. Chellappa. Fusion of Gait and Face for Human Identification. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 5, pages 901–904, Montreal, Canada, May 2004.

[105] E. J. C. Kelkboom, B. Gkberk, T. A. M. Kevenaar, A. H. M. Akkermans, and M. van der Veen. "3D Face": Biometric Template Protection for 3D Face Recognition. In *Proceedings of Second International Conference on Biometrics*, pages 566–573, Seoul, South Korea, August 2007.

[106] A. Kholmatov and B. Yanikoglu. Realization of Correlation Attack Against the Fuzzy Vault Scheme. In *Proceedings of SPIE Symposium on Security, Forensics, Steganography, and Watermarking of Multimedia Contents X (To appear)*, San Jose, USA, January 2008.

[107] T. Kinnunen, V. Hautamaki, and P. Franti. Fusion of Spectral Feature Sets for Accurate Speaker Identification. In *Ninth Conference on Speech and Computer*, pages 361–365, Saint-Petersburg, Russia, September 2004.

[108] J. Kittler, M. Hatef, R. P. Duin, and J. G. Matas. On Combining Classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, March 1998.

[109] J. Kittler and M. Sadeghi. Physics-based Decorrelation of Image Data for Decision Level Fusion in Face Verification. In *Fifth International Workshop on Multiple Classifier Systems*, pages 354–363, Cagliari, Italy, June 2004.

[110] D. V. Klien. Foiling the Cracker; A Survey of, and Improvements to Unix Password Security. In *Proceedings of the Second USENIX Workshop on Security*, pages 5–14, August 1990.

[111] S. Krawczyk and A. K. Jain. Securing Electronic Medical Records using Biometric Authentication. In *Proceedings of Fifth International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA)*, pages 1110–1119, Rye Brook, USA, July 2005.

[112] A. Kumar, D. C. M. Wong, H. C. Shen, and A. K. Jain. Personal Verification Using Palmprint and Hand Geometry Biometric. In *Fourth International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA)*, pages 668–678, Guildford, UK, June 2003.

[113] A. Kumar and D. Zhang. Personal Authentication using Multiple Palmprint Representation. *Pattern Recognition*, 38(10):1695–1704, October 2005.

[114] L. I. Kuncheva. *Combining Pattern Classifiers - Methods and Algorithms*. Wiley, 2004.

[115] RSA Laboratories. PKCS #5: Password-Based Crptography Standard, Version 2.0. Technical report, RSA Laboratories, March 1999.

[116] L. Lam and C. Y. Suen. Application of Majority Voting to Pattern Recognition: An Analysis of its Behavior and Performance. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, 27(5):553–568, 1997.

[117] Y. J. Lee, K. Bae, S. J. Lee, K. R. Park, and J. Kim. Biometric Key Binding: Fuzzy Vault based on Iris Images. In *Proceedings of Second International Conference on Biometrics*, pages 800–808, Seoul, South Korea, August 2007.

[118] E. L. Lehmann and J. P. Romano. *Testing Statistical Hypotheses*. Springer, 2005.

[119] J. Q. Li and A. Barron. Mixture Density Estimation. In S. A. Solla, T. K. Leen, and K.-R. Muller, editors, *Advances in Neural Information Processings Systems 12*. Morgan Kaufmann Publishers, San Mateo, USA, 1999.

[120] Q. Li and E. C. Chang. Robust, Short and Sensitive Authentication Tags Using Secure Sketch. In *Proceedings of ACM Multimedia and Security Workshop*, pages 56–61, Geneva, Switzerland, September 2006.

[121] Y. Li, S. Gong, and H. Liddell. Constructing Facial Identity Surfaces for Recognition. *International Journal of Computer Vision*, 53(1):71–92, June 2003.

[122] S. Lin and D. J. Costello. *Error Control Coding: Fundamentals and Applications*. Prentice Hall, Englewood Cliffs, USA, 1983.

[123] X. Liu and T. Chen. Geometry-assisted Statistical Modeling for Face Mosaicing. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, volume 2, pages 883–886, Barcelona, Spain, September 2003.

[124] X. Lu and A. K. Jain. Integrating Range and Texture Information for 3D Face Recognition. In *IEEE Computer Society Workshop on Application of Computer Vision (WACV)*, pages 156–163, Breckenridge, USA, January 2005.

[125] X. Lu, Y. Wang, and A. K. Jain. Combining Classifiers for Face Recognition. In *IEEE International Conference on Multimedia and Expo (ICME)*, volume 3, pages 13–16, Baltimore, USA, July 2003.

[126] L. Ma, T. Tan, Y. Wang, and D. Zhang. Personal Identification Based on Iris Texture Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1519–1533, December 2003.

[127] Y. Ma, B. Cukic, and H. Singh. A Classification Approach to Multi-biometric Score Fusion. In *Fifth International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA)*, pages 484–493, Rye Brook, USA, July 2005.

[128] D. Maio, D. Maltoni, J. L. Wayman, and A. K. Jain. FVC2002: Second Fingerprint Verification Competition. In *Proceedings of International Conference on Pattern Recognition (ICPR)*, pages 811–814, Quebec City, Canada, August 2002.

[129] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar. *Handbook of Fingerprint Recognition*. Springer-Verlag, 2003.

[130] G. L. Marcialis and F. Roli. Fingerprint Verification by Fusion of Optical and Capacitive Sensors. *Pattern Recognition Letters*, 25(11):1315–1322, August 2004.

[131] G. L. Marcialis and F. Roli. Fusion of Appearance-based Face Recognition Algorithms. *Pattern Analysis and Applications*, 7(2):151–163, July 2004.

[132] G. L. Marcialis and F. Roli. Fusion of Multiple Fingerprint Matchers by Single-layer Perceptron with Class-separation Loss Function. *Pattern Recognition Letters*, 26(12):1830–1839, September 2005.

[133] T. Matsumoto, M. Hirabayashi, and K. Sato. A Vulnerability Evaluation of Iris Matching (Part 3). In *Proceedings of the 2004 Symposium on Cryptography and Information Security*, pages 701–706, Iwate, Japan, January 2004.

[134] T. Matsumoto, H. Matsumoto, K. Yamada, and S. Hoshino. Impact of Artificial Gummy Fingers on Fingerprint Systems. In *Optical Security and Counterfeit Deterrence Techniques IV, Proceedings of SPIE*, volume 4677, pages 275–289, San Jose, USA, January 2002.

[135] O. Melnik, Y. Vardi, and C.-H. Zhang. Mixed Group Ranks: Preference and Confidence in Classifier Combination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(8):973–981, August 2004.

[136] K. D. Mitnick, W. L. Simon, and S. Wozniak. *The Art of Deception: Controlling the Human Element of Security*. Wiley, 2002.

[137] F. Monrose, M. K. Reiter, and S. Wetzel. Password Hardening Based on Keystroke Dynamics. In *Proceedings of Sixth ACM Conference on Computer and Communications Security*, pages 73–82, 1999.

[138] F. Monrose, M.K. Reiter, Q. Li, and S. Wetzel. Cryptographic Key Generation from Voice. In *Proceedings of IEEE Symposium on Security and Privacy*, pages 202–213, Oakland, USA, May 2001.

[139] H. Moon and P. J. Phillips. Computational and Performance Aspects of PCA-based Face Recognition Algorithms. *Perception*, 30(5):303–321, 2001.

[140] Y. S. Moon, H. W. Yeung, K. C. Chan, and S. O. Chan. Template Synthesis and Image Mosaicking for Fingerprint Registration: An Experimental Study. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 5, pages 409–412, Montreal, Canada, May 2004.

[141] A. Nagar and S. Chaudhury. Biometrics based Asymmetric Cryptosystem Design Using Modified Fuzzy Vault Scheme. In *Proceedings of IEEE International Conference Pattern Recognition*, volume 4, pages 537–540, Hong Kong, China, August 2006.

[142] K. Nandakumar, Y. Chen, S. C. Dass, and A. K. Jain. Likelihood Ratio Based Biometric Score Fusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):342–347, February 2008.

[143] K. Nandakumar, Y. Chen, A. K. Jain, and S. C. Dass. Quality-based Score Level Fusion in Multibiometric Systems. In *Proceedings of International Conference on Pattern Recognition (ICPR)*, pages 473–476, Hong Kong, China, August 2006.

[144] K. Nandakumar, A. K. Jain, and S. Pankanti. Fingerprint-based Fuzzy Vault: Implementation and Performance. *IEEE Transactions on Information Forensics and Security*, 2(4):744–757, December 2007.

[145] K. Nandakumar, A. Nagar, and A. K. Jain. Hardening Fingerprint Fuzzy Vault Using Password. In *Proceedings of Second International Conference on Biometrics*, pages 927–937, Seoul, South Korea, August 2007.

[146] R. B. Nelsen. *An Introduction to Copulas*. Springer, 1999.

[147] BBC News. Long Lashes Thwart ID Scan Trial. Available at `http://news.bbc.co.uk/2/hi/uk_news/politics/3693375.stm`, May 2004.

[148] Biometric System Laboratory University of Bologna. FVC2006: The Fourth International Fingerprint Verification Competition. Available at `http://bias.csr.unibo.it/fvc2006/default.asp`.

[149] Department of Homeland Security. Privacy Impact Assessment for the Automated Biometric Identification System (IDENT). Available at `http://www.dhs.gov/xlibrary/assets/privacy/privacy_pia_usvisit_ident_final.pdf`, July 2006.

[150] Federal Bureau of Investigation. Integrated Automated Fingerprint Identification System. Available at `http://www.fbi.gov/hq/cjisd/iafis.htm`.

[151] National Institute of Standards and Technology. NIST Biometric Scores Set. Available at `http://http://www.itl.nist.gov/iad/894.03/biometricscores`, 2004.

[152] P. J. Phillips, K. W. Bowyer, and P. J. Flynn. Comments on the CASIA Version 1.0 Iris Data Set. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(10):1869–1870, October 2007.

[153] P. J. Phillips, W. T. Scruggs, A. J. OToole, P. J. Flynn, K. W. Bowyer, C. L. Schott, and M. Sharpe. FRVT 2006 and ICE 2006 Large-Scale Results. Technical Report NISTIR 7408, NIST, March 2007.

[154] N. Poh and S. Bengio. Database, Protocol and Tools for Evaluating Score-Level Fusion Algorithms in Biometric Authentication. *Pattern Recognition*, 39(2):223–233, February 2006.

[155] S. Prabhakar and A. K. Jain. Decision-level Fusion in Fingerprint Verification. *Pattern Recognition*, 35(4):861–874, April 2002.

[156] M. Przybocki and A. Martin. NIST Speaker Recognition Evaluation Chronicles. In *Odyssey: The Speaker and Language Recognition Workshop*, pages 12–22, Toledo, Spain, May 2004.

[157] A. Rakhlin, D. Panchenko, and S. Mukherjee. Risk Bounds for Mixture Density Estimation. *ESAIM: Probability and Statistics*, 9:220–229, June 2005.

[158] N. K. Ratha, S. Chikkerur, J. H. Connell, and R. M. Bolle. Generating Cancelable Fingerprint Templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4):561–572, April 2007.

[159] N. K. Ratha, J. H. Connell, and R. M. Bolle. Image Mosaicing For Rolled Fingerprint Construction. In *Proceedings of Fourteenth International Conference on Pattern Recognition (ICPR)*, volume 2, pages 1651–1653, Brisbane, Australia, August 1998.

[160] N. K. Ratha, J. H. Connell, and R. M. Bolle. An Analysis of Minutiae Matching Strength. In *Proceedings of Third International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pages 223–228, Halmstad, Sweden, June 2001.

[161] N. K. Ratha, J. H. Connell, R. M. Bolle, and S. Chikkerur. Cancelable Biometrics: A Case Study in Fingerprints. In *Proceedings of IEEE International Conference Pattern Recognition*, volume 4, pages 370–373, Hong Kong, China, August 2006.

[162] D. Reynolds, W. Andrews, J. Campbell, J. Navratil, B. Peskin, A. Adami, Q. Jin, D. Klusacek, J. Abramson, R. Mihaescu, J. Godfrey, D. Jones, and B. Xiang. The SuperSID Project: Exploiting High-level Information for High-accuracy Speaker Recognition. In *IEEE International Conference on Acoustics,*

*Speech, and Signal Processing (ICASSP)*, pages 784–787, Hong Kong, China, April 2003.

[163] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker Verification using Adapted Gaussian Mixture Models. *Digital Signal Processing*, 10:19–41, January/April/July 2000.

[164] J. A. Rice. *Mathematical Statistics and Data Analysis, Second Edition*. Duxbury Press, 1995.

[165] L. Rodriguez-Linares, C. Garcia-Mateo, and J. L. Alba-Castro. On Combining Classifiers for Speaker Authentication. *Pattern Recognition*, 36(2):347–359, February 2003.

[166] A. Ross and R. Govindarajan. Feature Level Fusion Using Hand and Face Biometrics. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification II*, volume 5779, pages 196–204, Orlando, USA, March 2005.

[167] A. Ross and A. K. Jain. Information Fusion in Biometrics. *Pattern Recognition Letters*, 24(13):2115–2125, September 2003.

[168] A. Ross, A. K. Jain, and J. Reisman. A Hybrid Fingerprint Matcher. *Pattern Recognition*, 36(7):1661–1673, July 2003.

[169] A. Ross, K. Nandakumar, and A. K. Jain. *Handbook of Multibiometrics*. Springer, 2006.

[170] A. Ross, S. Shah, and J. Shah. Image Versus Feature Mosaicing: A Case Study in Fingerprints. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification*, volume 6202, pages 1–12, Orlando, USA, April 2006.

[171] A. K. Ross, J. Shah, and A. K. Jain. From Templates to Images: Reconstructing Fingerprints From Minutiae Points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4):544–560, 2007.

[172] C. Sanderson and K. K. Paliwal. Information Fusion for Robust Speaker Verification. In *Seventh European Conference on Speech Communication and Technology*, pages 755–758, Aalborg, Denmark, September 2001.

[173] C. Sanderson and K. K. Paliwal. Information Fusion and Person Verification Using Speech and Face Information. Technical Report IDIAP-RR 02-33, IDIAP, September 2002.

[174] M. Savvides and B. V. K. Vijaya Kumar. Cancellable Biometric Filters for Face Recognition. In *Proceedings of IEEE International Conference Pattern Recognition*, volume 3, pages 922–925, Cambridge, UK, August 2004.

[175] W. J. Scheirer and T. E. Boult. Cracking Fuzzy Vaults and Biometric Encryption. In *Proceedings of Biometrics Symposium*, September 2007.

[176] Luchthaven Schiphol. Privium: A Select Way to Travel. Available at `http://www.schiphol.nl/privium/privium.jsp`.

[177] S. Shah. Enhanced Iris Recognition: Algorithms for Segmentation, Matching and Synthesis. Master's thesis, Lane Department of Computer Science and Electrical Engineering, West Virginia University, 2006.

[178] G. Shakhnarovich, L. Lee, and T.J. Darrell. Integrated Face and Gait Recognition from Multiple Views. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 439–446, Hawaii, USA, December 2001.

[179] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall, 1986.

[180] R. Snelick, U. Uludag, A. Mink, M. Indovina, and A. K. Jain. Large Scale Evaluation of Multimodal Biometric Authentication Using State-of-the-Art Systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):450–455, March 2005.

[181] D. A. Socolinsky, A. Selinger, and J. D. Neuheisel. Face Recognition with Visible and Thermal Infrared Imagery. *Computer Vision and Image Understanding*, 91(1-2):72–114, July-August 2003.

[182] B. Son and Y. Lee. Biometric Authentication System Using Reduced Joint Feature Vector of Iris and Face. In *Proceedings of Fifth International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pages 513–522, Rye Brook, USA, July 2005.

[183] O. T. Song, A. B. J Teoh, and D. C. L. Ngo. Application-Specific Key Release Scheme from Biometrics. *International Journal of Network Security*, 6(2):127–133, March 2008.

[184] C. Soutar, D. Roberge, A. Stoianov, R. Gilroy, and B. V. K. V. Kumar. Biometric Encrpytion. In R. K. Nichols, editor, *ICSA Guide to Cryptography*. McGraw Hill, 1999.

[185] Y. Sutcu, Q. Li, and N. Memon. Protecting Biometric Templates with Sketch: Theory and Practice. *IEEE Transactions on Information Forensics and Security*, 2(3):503–512, September 2007.

[186] Y. Sutcu, Q. Li, and N. Memon. Secure Biometric Templates from Fingerprint-Face Features. In *Proceedings of CVPR Workshop on Biometrics*, Minneapolis, USA, June 2007.

[187] A. B. J. Teoh, A. Goh, and D. C. L. Ngo. Random Multispace Quantization as an Analytic Mechanism for BioHashing of Biometric and Random Identity Inputs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):1892–1901, December 2006.

[188] A. B. J. Teoh, K.-A. Toh, and W. K. Yip. $2^N$ Discretisation of BioPhasor in Cancellable Biometrics. In *Proceedings of Second International Conference on Biometrics*, pages 435–444, Seoul, South Korea, August 2007.

[189] NIST Report to the United States Congress. Summary of NIST Standards for Biometric Accuracy, Tamper Resistance, and Interoperability. Available at `ftp://sequoyah.nist.gov/pub/nist_internal_reports/NISTAPP_Nov02.pdf`, November 2002.

[190] K.-A. Toh, X. Jiang, and W.-Y. Yau. Exploiting Global and Local Decisions for Multimodal Biometrics Verification. *IEEE Transactions on Signal Processing, (Supplement on Secure Media)*, 52(10):3059–3072, October 2004.

[191] K.-A. Toh, W. Xiong, W.-Y. Yau, and X. Jiang. Combining Fingerprint and Hand-Geometry Verification Decisions. In *Fourth International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA)*, pages 688–696, Guildford, UK, June 2003.

[192] K.-A. Toh and W.-Y. Yau. Fingerprint and Speaker Verification Decisions Fusion Using a Functional Link Network. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Applications and Reviews*, 35(3):357–370, August 2005.

[193] M. Turk and A. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[194] P. Tuyls, A. H. M. Akkermans, T. A. M. Kevenaar, G.-J. Schrijen, A. M. Bazen, and R. N. J. Veldhuis. Practical Biometric Authentication with Template Protection. In *Proceedings of Fifth International Conference on Audio- and Video-based Biometric Person Authentication*, pages 436–446, Rye Town, USA, July 2005.

[195] B. Ulery, A. R. Hicklin, C. Watson, W. Fellner, and P. Hallinan. Studies of Biometric Fusion. Technical Report IR 7346, NIST, September 2006.

[196] U. Uludag and A. K. Jain. Securing Fingerprint Template: Fuzzy Vault With Helper Data. In *Proceedings of CVPR Workshop on Privacy Research In Vision*, page 163, New York, USA, June 2006.

[197] U. Uludag, S. Pankanti, and A. K. Jain. Fuzzy Vault for Fingerprints. In *Proceedings of Fifth International Conference on Audio- and Video-based Biometric Person Authentication*, pages 310–319, Rye Town, USA, July 2005.

[198] U. Uludag, S. Pankanti, S. Prabhakar, and A. K. Jain. Biometric Cryptosystems: Issues and Challenges. *Proceedings of the IEEE, Special Issue on Multimedia Security for Digital Rights Management*, 92(6):948–960, June 2004.

[199] A. Vetro and N. Memon. Biometric System Security. Tutorial presented at Second International Conference on Biometrics, Seoul, South Korea, August 2007.

[200] C. Vielhauer, R. Steinmetz, and A. Mayerhofer. Biometric Hash Based on Statistical Features of Online Signatures. In *Proceedings of* $16^{th}$ *International Conference on Pattern Recognition*, volume 1, pages 123–126, Quebec, Canada, August 2002.

[201] A. Wald. Sequential Tests of Statistical Hypotheses. *The Annals of Mathematical Statistics*, 16(2):117–186, June 1945.

[202] M. P. Wand and M. C Jones. *Kernel Smoothing*. Chapman & Hall, CRC Press, 1995.

[203] Y. Wang, T. Tan, and A. K. Jain. Combining Face and Iris Biometrics for Identity Verification. In *Fourth International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA)*, pages 805–813, Guildford, UK, June 2003.

[204] 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. Technical Report NISTIR 7123, NIST, June 2004.

[205] K. Woods, K. Bowyer, and W. P. Kegelmeyer. Combination of Multiple Classifiers Using Local Accuracy Estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):405–410, April 1997.

[206] L. Xu, A. Krzyzak, and C. Y. Suen. Methods for Combining Multiple Classifiers and their Applications to Handwriting Recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(3):418–435, 1992.

[207] F. Yang, M. Paindavoine, H. Abdi, and A. Monopoli. Development of a Fast Panoramic Face Mosaicking and Recognition System. *Optical Engineering*, 44(8), August 2005.

[208] J. Yang, J.-Y. Yang, D. Zhang, and J.-F. Lu. Feature Fusion: Parallel Strategy vs. Serial Strategy. *Pattern Recognition*, 38(6):1369–1381, June 2003.

[209] S. Yang and I. Verbauwhede. Automatic Secure Fingerprint Verification System Based on Fuzzy Vault Scheme. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 609–612, Philadelphia, USA, March 2005.

[210] B. Yanikoglu and A. Kholmatov. Combining Multiple Biometrics to Protect Privacy. In *Proceedings of ICPR Workshop on Biometrics: Challenges arising from Theory to Practice*, Cambridge, UK, August 2004.

[211] J. You, W.-K. Kong, D. Zhang, and K. H. Cheung. On Hierarchical Palmprint Coding With Multiple Features for Personal Identification in Large Databases. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(2):234–243, February 2004.

[212] Y.-L. Zhang, J. Yang, and H. Wu. A Hybrid Swipe Fingerprint Mosaicing Scheme. In *Proceedings of Fifth International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pages 131–140, Rye Brook, USA, July 2005.

[213] S. Zhou, V. Krueger, and R. Chellappa. Probabilistic Recognition of Human Faces from Video. *Computer Vision and Image Understanding*, 91(1-2):214–245, July-August 2003.

[214] X. Zhou. Template Protection and its Implementation in 3D Face Recognition Systems. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification*, volume 6539, pages 214–225, Orlando, USA, April 2007.

[215] Y. Zhu, S. C. Dass, and Jain. Statistical Models for Assessing the Individuality of Fingerprints. *IEEE Transactions on Information Forensics and Security*, 2(3):391–401, September 2007.

[216] Z. Zivkovic and F. van der Heijden. Recursive Unsupervised Learning of Finite Mixture Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(5):651–656, May 2004.