# Fingerprint Classification and Matching Using a Filterbank

By

*Salil Prabhakar*

A DISSERTATION

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

DOCTOR OF PHILOSOPHY

Computer Science & Engineering

2001

# Abstract

## Fingerprint Classification and Matching Using a Filterbank

By

*Salil Prabhakar*

Accurate automatic personal identification is critical in a variety of applications in our electronically interconnected society. Biometrics, which refers to identification based on physical or behavioral characteristics, is being increasingly adopted to provide positive identification with a high degree of confidence. Among all the biometric techniques, fingerprint-based authentication systems have received the most attention because of the long history of fingerprints and their extensive use in forensics. However, the numerous fingerprint systems currently available still do not meet the stringent performance requirements of several important civilian applications. To assess the performance limitations of popular minutiae-based fingerprint verification system, we theoretically estimate the probability of a false correspondence between two fingerprints from different fingers based on the minutiae representation of fingerprints. Due to the limited amount of information present in the minutiae-based representation, it is desirable to explore alternative representations of fingerprints.

We present a novel filterbank-based representation of fingerprints. We have used this compact representation for fingerprint classification as well as fingerprint verification. Experimental results show that this algorithm competes well with the state-of-the-art minutiae-based matchers. We have developed a decision level information fusion framework which improves the fingerprint verification accuracy when multiple matchers, multiple fingers of the user, or multiple impressions of the same finger are combined. A feature verification and purification scheme is proposed to improve the performance of the minutiae-based matcher.

To My Family

# Acknowledgments

During my four years of studies at Michigan State University, the sheer joy of working with my advisor, Dr. Anil K. Jain by far exceeded the excitement of working in pattern recognition, the sense of achievement on completing a Ph.D. thesis, or watching the school win a NCAA basketball championship. His love for perfection and interest in detail have supplemented my own quest for knowledge. His advise, guidance, help, ideas, insights, encouragement, regular reminders of "keep working hard" and enquiries of "any new breakthroughs?" were instrumental in making this thesis possible and shaped my research career. I would like to thank Dr. S. Pankanti of IBM T. J. Watson Research Center for numerous discussions, suggestions, insights, and help, Dr. G. Stockman and Dr. J. Zacks for serving on my Ph.D. committee, Dr. J. Weng for useful discussions, and Dr. R. Bolle, Manager, Exploratory Computer Vision Group, IBM T. J. Watson Research Center for his support. I would like to especially thank my mentor Dr. Lin Hong for his help during my first year of graduate studies.

Special thanks to Arun Ross, Scott Connell, Aditya Vailaya, Nico Duta, Shaoyun Chen, Wey Hwang, Yonghong Li, Vera Bakic, Paul Albee, Anoop Namboodri, Erin

McGarrity, Vincent Hsu, Dan Gutchess, Friederike Griess, Yatin Kulkarni, and others in the PRIP lab for numerous discussions and encouragement.

I would also like to thank Cathy Davison, Starr Portice, Linda Moore, Debbie Kruch, Beverly J. Wallace, and Karen Lillis for their administrative help.

My sincere thanks go to my parents for their never-fading love and encouragement, and to my wife, Chandini, for her understanding and love.

# Table of Contents

**LIST OF TABLES** **x**

**LIST OF FIGURES** **xiii**

**1 Introduction** **1**
1.1 Automatic Identification . . . . . 1
1.2 Biometrics . . . . . 3
1.3 Applications . . . . . 4
1.4 Fingerprints . . . . . 5
1.5 Fingerprint Formation . . . . . 6
1.6 Fingerprint Individuality . . . . . 7
1.7 Fingerprint Sensors . . . . . 10
1.8 Fingerprint Representation . . . . . 14
1.9 Fingerprint Classification . . . . . 16
1.10 Fingerprint Verification . . . . . 19
1.11 Information Fusion . . . . . 22
1.12 Feature Verification . . . . . 23
1.13 Challenges in Automatic Fingerprint Identification . . . . . 24
1.14 State-of-the-art in Fingerprint Identification . . . . . 26
1.15 Thesis Objectives . . . . . 27
1.16 Thesis Outline . . . . . 29

**2 On the Individuality of Fingerprints** **30**
2.1 Genetic Factors . . . . . 32
2.1.1 Introduction . . . . . 32
2.1.2 Experimental Results . . . . . 38
2.1.3 Summary . . . . . 45
2.2 Environmental Factors . . . . . 47
2.2.1 Introduction . . . . . 47
2.2.2 Background . . . . . 51
2.2.3 A Model of Fingerprint Individuality . . . . . 62
2.2.4 Experimental Results and Discussions . . . . . 78
2.2.5 Summary . . . . . 84

**3 Fingerprint as Oriented Texture** **88**
3.1 Introduction . . . . . 89
3.2 Reference Point Location . . . . . 96
3.3 Tessellation . . . . . 102

3.4 Filtering . . . 105
3.5 Feature Vector . . . 112
3.6 Summary . . . 116

**4 Fingerprint Classification** **119**
4.1 Introduction . . . 121
4.2 Feature Extraction . . . 126
4.3 Classification . . . 130
4.4 Experimental Results . . . 132
4.4.1 Dataset . . . 132
4.4.2 $K$-Nearest neighbor classifier . . . 135
4.4.3 Neural network classifier . . . 136
4.4.4 Two-stage classifier . . . 137
4.4.5 Reject option . . . 141
4.4.6 Support vector machine classifier . . . 142
4.4.7 Consistency results . . . 145
4.4.8 Defining New Classes . . . 145
4.4.9 Dimensionality Reduction Using PCA . . . 147
4.4.10 Dimensionality Reduction Using Feature Clustering . . . 148
4.5 Summary . . . 150

**5 Fingerprint Matching** **152**
5.1 Introduction . . . 156
5.2 Feature Extraction . . . 156
5.3 Matching . . . 160
5.4 Experimental Results . . . 162
5.5 Summary . . . 172

**6 Decision-level Fusion in Fingerprint Verification** **176**
6.1 Introduction . . . 177
6.2 Matcher Combination . . . 179
6.3 Integration Strategy . . . 181
6.3.1 Matcher Selection . . . 181
6.3.2 Non-parametric density estimation . . . 182
6.3.3 Decision Strategy . . . 183
6.4 Matching Algorithms . . . 184
6.4.1 Hough Transform Based Matching (Algorithm *Hough*) . . . 185
6.4.2 String Distance Based Matching (Algorithm *String*) . . . 185
6.4.3 2D Dynamic Programming Based Matching (Algorithm *Dynamic*) . . . 186
6.4.4 Filterbank Based Matching (Algorithm *Filter*) . . . 187
6.5 Experimental Results . . . 187
6.6 Summary . . . 203

**7 Fingerprint Feature Detection and Verification** **204**
7.1 Introduction . . . . . 205
7.2 Minutia Verification . . . . . 207
7.2.1 Feature Extraction . . . . . 207
7.2.2 Training . . . . . 208
7.2.3 Testing . . . . . 210
7.3 Minutia Classification . . . . . 210
7.4 Experimental Results . . . . . 213
7.5 Summary . . . . . 215

**8 Conclusions and Future Work** **217**
8.1 Conclusions and Research Contributions . . . . . 217
8.2 Future Directions . . . . . 222

**BIBLIOGRAPHY** **227**

# List of Tables


1.1 Performance of fingerprint verification systems reported by various companies on their web sites. None of the companies mention the database used for obtaining the performance results, and thus the performance numbers can not be directly compared. FAR: False Accept Rate; FRR: False Reject Rate. . . . . 27

1.2 Comparison of state-of-the-art fingerprint verification algorithms in terms of equal error rate (ERR) and timing on a database of 800 fingerprints (image size = $448 \times 478$ captured by DF-90 optical sensor manufactured by Identicator Technology). Details of the evaluation protocol can be found in [57]. . . . . 28

2.1 False accept and false reject rates with different threshold values for the twin database. . . . . 43

2.2 Fingerprint features used in different models. . . . . 56

2.3 Comparison of probability of a particular fingerprint configuration using different models. For a fair comparison, we do not distinguish between minutiae types. By assuming that an average size fingerprint has 24 regions ($R = 24$) as defined by Galton, 72 regions ($M = 72$) as defined by Osterburg et al., and has 36 minutiae on an average ($N = 36$), we compare the probability of observing a given fingerprint configuration in the third column of the table. The probability of observing a fingerprint configuration with $N = 12$, and equivalently, $R = 8$, is given in braces in the third column. Note that all probabilities represent a full ($N$ minutiae) match as opposed to a partial match (see Table 2.5). 57

2.4 The effects of the fingerprint expert misjudgments in using the 12-point rule. The source of error could be in underestimating the minutiae detected in the latent print ($n$) or overestimating the correct number of matched minutiae ($q$). $m = 12$ for all entries. Except for $(m = 12, n = 12, q = 12)$ entry, all other entries represent incorrect judgments by the fingerprint expert. For instance, the entry $(m = 12, n = 14, q = 8)$ in the table indicates that although the fingerprint examiner determined that 12 template minutia unequivocally matched with all 12 input minutiae, there were indeed 14 input minutiae (2 missed input minutiae) out of which only 8 correctly matched with the corresponding template minutiae (4 incorrect match judgments). 82

2.5 Fingerprint correspondence probabilities obtained from the proposed individuality model for different sizes of fingerprint images containing 26, 36 or 46 minutiae. $M$ for the last entry was computed by estimating typical print area manifesting 12 minutia in a 500 dpi optical fingerprint scan. The entry $(35, 12, 12, 12)$ corresponds to the 12-point rule. . . . . 83

2.6 Fingerprint correspondence probabilities obtained from matching imposter fingerprints using an AFMS [11] for the MSU_VERIDICOM and MSU_DBI databases. The probabilities given in the table are for matching "exactly $q$" minutiae. The probabilities for matching "$q$ or more" minutiae are $3.0\times10^{-2}$ and $3.2\times10^{-2}$ for the MSU_VERIDICOM and MSU_DBI databases, respectively, i.e., of the same order. The average values for $M$, $m$, and $n$ are $28,383$, 26, and 26 for the MSU_VERIDICOM database and $67,415$, 46 and 46 for the MSU_DBI database, respectively. . . . . 84

3.1 Gabor filter mask of size $33 \times 33$, $\theta = 0°$, $f = 0.1$, $\delta_x = \delta_y = 4.0$. Only a $19 \times 19$ matrix from the center of the $33 \times 33$ filter is shown because the mask values outside this are zero. Also, only the top left quarter of the mask is shown due to the symmetry in the X and Y axes of the $0°$ oriented filter. The mask values less than 0.05 are set to zero. Each entry is to be multiplied by $10^{-3}$. . . . . 111

4.1 Fingerprint classification literature survey. The number of classes is denoted by $C$, the classification accuracy is denoted by $Acc$, and the reject rate is denoted by $RR$. The classification accuracies reported by the different authors are on different databases with different number of fingerprints and therefore, they cannot be directly compared. Most of the work in fingerprint classification is based on supervised learning and discrete class assignment using knowledge-based features. . . . . 123

4.2 Confusion matrix for the $K$-nearest neighbor classification for the five-class problem; $K = 10$. . . . . 135

4.3 Confusion matrix for the $K$-nearest neighbor classification for the four-class problem; $K = 10$. . . . . 137

4.4 Confusion matrix for the neural network classification for the five-class problem. . . . . 137

4.5 Confusion matrix for the neural network classification for the four-class problem. . . . . 138

4.6 Confusion matrix for the two-stage classification for the five-class problem. 138

4.7 Confusion matrix for the two-stage classification for the four-class problem. 139

4.8 Error-reject tradeoff. . . . . 142

4.9 A comparison of various fingerprint classification algorithms on the NIST 4 database. . . . . 144

5.1 Fingerprint matcher literature survey. The fingerprint matching algorithms are classified based on the alignment assumed between the template and the input fingerprint features. The rotation is denoted by $R$, the translation is denoted by $T$, and the scale is denoted by $S$. . . . . 155
5.2 False acceptance and false reject rates with different threshold values for the MSU_DBI database. . . . . 168
5.3 Comparison of the equal error rates (ERR) of the proposed filterbank-based technique with a state-of-the-art minutiae-based technique on two different databases. . . . . 172

6.1 Confidence-level classifier combination schemes. A more detailed comparison can be found in [15]. . . . . 180
6.2 Combining two fingerprint matchers. $CS$ is the class separation statistic. $CS$ and $\rho$ are computed from the training data. Ranks by $EER$ (Equal Error Rate) are computed from the independent test data. . . . . 192
6.3 Comparison of the performance of the best matcher combination with the best individual matcher. GAR refers to the genuine acceptance rate that is plotted on the ordinate of the $ROC$ curves. We performed ten runs of the combination scheme with ten different splits of the database into training and test sets. The mean ($Mean$) and variance ($Var$) of the GAR values for three fixed values of FAR are reported. . . . . 198
6.4 Equal error rate improvement due to combination of matchers. . . . . 201

# List of Figures


1.1 Various electronic access applications in widespread use that require automatic authentication. . . . . 2

1.2 Orientation field, thinned ridges, minutiae, and singular points. . . . . 3

1.3 Fingerprint images captured using (a) inked method (NIST-9 database), image size = 832 × 768 pixels, (b) Digital Biometrics optical sensor (MSU_DBI database), image size = 508×480 pixels, and (c) Veridicom solid-state sensor (MSU_VERIDICOM database), image size = 300 × 300 pixels. All the images have 256 gray levels. . . . . 12

1.4 Fingerprint sensors. (a) Optical sensor from Digital Biometrics, Inc., and (b) solid-state sensor from Veridicom, Inc. . . . . 13

1.5 Six major fingerprint classes. Twin loop images are labeled as whorl in the NIST-4 database. . . . . 17

1.6 System diagram for an automatic verification system. . . . . 19

1.7 A general pattern recognition system with proposed feedback in feature extraction and a new feature refinement stage. . . . . 24

1.8 An example fingerprint image from the NIST-4 database. The experts have labeled this image to belong to two classes, right loop, and tented arch. . . . . 25

2.1 Photograph of identical twin sisters (www.visi.com/∼charlesr/). . . . . 33

2.2 Fingerprint images of identical twin sisters captured using an optical scanner from Digital Biometrics Inc., (a) and (b) are two impressions of the same finger of one twin and (c) and (d) are two impressions of the corresponding finger of her sibling. Matching score between (a) and (b) is 487, and between (c) and (d) is 510. The matching score between (a) and (c) is 24, and the matching score between (b) and (d) is 4. The fingerprints of both the twins here have the same type (right loop) and look similar to untrained eyes. Fingerprint experts, as well as our automatic fingerprint identification system can, however, easily differentiate the twins. . . . . 34

2.3 Minutiae extraction for twins. (a) and (b) are fingerprint images of an identical twin and his/her sibling while the fingerprint in (c) is from another person. (d), (e), and (f) are the minutiae extracted from (a), (b), and (c), respectively using the extraction algorithm in [11]. . . . 36

2.4 Minutiae matching for (a) twin-nontwin (matching of Figures 2.3(e) and 2.3(f), matching score = 3 on a scale of 0-999) and (b) twin-twin (matching of Figures 2.3(d) and Figure 2.3(e), matching score = 38 on a scale of 0-999). The "matched" minutiae pairs are shown by bounding boxes. . . . . 37
2.5 Minutiae matching for two impressions of the same finger shown in Figures 2.2(a) and 2.2(b) (matching score = 487 on a scale of 0-999). The "matched" minutiae pairs are shown by bounding boxes. . . . . 37
2.6 (a) Distribution of matching scores for twin-twin imposter, twin-nontwin imposter, and genuine fingerprint matchings. (b) ROC curves for twin-twin and twin-nontwin minutiae pattern matchings. . . . . 40
2.7 Effect of fingerprint class type on the matching score. . . . . 41
2.8 A fingerprint image of type "right loop". The overall ridge structure, singular points, and sweat pores are shown. . . . . 48
2.9 Automatic minutiae matching. Two impressions of the same finger were matched in (a) 39 minutiae were detected in input (left), 42 in template (right), and 36 "true" correspondences were found. Two different fingers are matched in (b) 64 minutiae were detected in input (left), 65 in template (right), and 25 "false" correspondences were found. . . 66
2.10 Fingerprint and minutiae. . . . . 67
2.11 Distribution of minutiae distance differences for the genuine fingerprint pairs in the *GT* database. . . . . 73
2.12 Distributions for minutiae angle differences for the (a) genuine fingerprint pairs using the ground truth and (b) imposter matchings using the automatic fingerprint matching system. . . . . 75
2.13 Area of overlap between the two fingerprints that are matched based on the bounding boxes of the minutiae features for (a) MSU_DBI database; (b) MSU_VERIDICOM database. . . . . 76
2.14 Distributions for $m$, $n$, and $q$ for computation of averages for (a) MSU_DBI database; (b) MSU_VERIDICOM database. . . . . 79
2.15 Comparison of experimental and theoretical probabilities for the number of matching minutiae. (a) MSU_DBI database; (b) MSU_VERIDICOM database. . . . . 80

3.1 Flow pattern in a fingerprint image. (a) A section of a fingerprint image, (b) 3-dimensional surface plot of (a). . . . . 90
3.2 Difficulty in fingerprint matching. (a) and (b) have the same global configuration but are images of two different fingers. . . . . 92
3.3 Schematic diagram for extraction of generic texture-based representation for fingerprints. . . . . 93
3.4 Fingerprint of (a) a child, and (b) an adult. Both the fingerprints were scanned at 500 *dpi*. . . . . 94
3.5 Concave and convex ridges in a fingerprint image when the finger is positioned upright. The reference point is marked by $X$. . . . . 96

3.6 Estimating the reference point. (a) Smoothed orientation field overlapped on the original image, (b) orientation field ($w$=10) shown as intensity distribution; the background has been segmented, and (c) *sine* component of the orientation field; the darkest pixel in the center of the image marks the detected reference point. Images have been scaled to the range 0-255 for viewing. . . . . 99
3.7 Regions for integrating pixel intensities in $\mathcal{E}$ for computing $\mathcal{A}(i,j)$. . . . . 101
3.8 Examples of the results of our reference point location algorithm. The algorithm fails on very poor quality fingerprints such as (c) and (d). . 103
3.9 Reference point ($\times$), the region of interest, and 80 sectors ($B = 5$, $k = 16$) superimposed on a fingerprint. . . . . 106
3.10 Fingerprints have well defined local frequency and orientation. Ridges in local regions are shown in (a) and (b). Fourier spectrum of (a) and (b) are shown in (c) and (d), respectively. . . . . 107
3.11 Gabor filters (mask size = $33 \times 33$, $f = 0.1$, $\delta_x = 4.0$, $\delta_y = 4.0$). Only 0° and 90° oriented filters are shown here. . . . . 108
3.12 Normalized, filtered, and reconstructed fingerprint images. (a) area of interest, (b) normalized image, (c)-(j) 0°, 22.5°, 45°, 90°, 112.5°, 157.5° filtered images, respectively, (k) reconstructed image with 4 filters, and (l) reconstructed image with 8 filters. While four filter orientations are sufficient to capture the global structure of the fingerprint, eight filter orientations are required to capture the local characteristics. . . . . 110
3.13 Examples of 640-dimensional feature vectors. (a) First impression of finger 1, (b) Second impression of finger 1, (c) and (d) are the corresponding FingerCodes, (e) First impression of finger 2, (f) Second impression of finger 2, (g) and (h) are the corresponding FingerCodes. . . . . 114
3.14 Example of new touchless fingerprint sensor TFS 050 from Biometric Partners, Inc. (http://www.biometricpartners.com/). The touchless sensor captures a fingerprint from a distance of approximately 50mm. Advantages of touchless technology include capture of larger fingerprint area, is more hygienic, the sensor does not degrade with repeated use, and there is no nonlinear distortion due to finger pressure difference in the captured image. The image captured by the sensor in (a) is shown in (b). However, the touchless sensors have their own problems, including poor quality images. . . . . 118

4.1 Pattern area and typelines [68, 104]. . . . . 124
4.2 Flow diagram of our fingerprint classification algorithm. . . . . 125
4.3 Reference point detected by the algorithm described in Chapter 3 ($\square$), moved reference point ($\times$), the region of interest and 48 sectors. . . . 127
4.4 Normalized, filtered, and reconstructed fingerprint images. . . . . 128
4.5 Reconstructed fingerprint images using (a) four filters, and (b) eight filters. Most of the directionality information is captured by four filters. . . 129

4.6 Fingerprint representation using 192-dimensional feature vectors (In each representation, the top left disc represents the 0° component, the top right disc represents the 45° component, the bottom left disc represents the 90° component, and the bottom right disc represents the 135° component). The test image is a right loop. Each disk corresponds to one particular filter and there are 48 features (shown as gray values) in each disk ($8 \times 6 = 48$ sectors) for a total of 192 ($48 \times 4$) features. . . 130
4.7 Two-stage classification scheme using K-NN and neural network classifiers. 131
4.8 Example of images in the NIST 4 database with two ground truth labels. The poor quality fingerprint in (a) is labeled as belonging to both the arch and tented arch classes, (b) is labeled as belonging to both the left loop and tented arch classes. . . . . 132
4.9 Example of images which were rejected because a valid tessellation could not be established. . . . . 133
4.10 $K$ vs. classification error for the $K$-nearest neighbor classifier for the five-class problem. . . . . 136
4.11 Poor quality images which were correctly classified. . . . . 139
4.12 Poor quality images which were misclassified as arch. . . . . 140
4.13 Misclassification of whorl (twin loop) as (a) right loop (b) left loop. . . . 141
4.14 Examples of arch-loop misclassifications; (a) a right loop misclassified as an arch; (b) an arch misclassified as a tented arch. . . . . 142
4.15 Examples of images rejected by $(10, 5)$-NN classifier. . . . . 143

5.1 System diagram of our fingerprint authentication system. . . . . 157
5.2 Examples of 640-dimensional feature vectors corresponding to nine different impressions of the same finger. . . . . 159
5.3 The fingerprint image in (b) is obtained by a $-22.5°$ rotation of (a). A part of the feature vector corresponding to the 0° Gabor filtered image extracted from (a) is shown in (c) as a gray scale image. The feature vector in (c) is rotated by $-22.5°$ ($R = -1$ in Equations (5.2) and (5.3)) and is shown in (d). (e) shows the feature vector extracted from the fingerprint image in (b). The feature vectors shown in (d) and (e) are similar illustrating that the feature vector for a $-22.5°$ rotation in the original image approximately corresponds to a unit anticlockwise cyclic rotation of the feature vector. . . . . 161
5.4 A comparison of the quality of inked fingerprints and dab fingerprints. (a) inked fingerprint, (b) dab fingerprint. . . . . 163
5.5 Examples of images with large deformation due to finger pressure differences in the MSU_DBI database. Fingerprint images in (b) and (d) were taken six weeks after the images in (a) and (c) were acquired, respectively. . . . . 164
5.6 Examples of rejected images. (a) a poor quality image, (b) the reference point is (correctly) detected at a corner of the image and so an appropriate region of interest could not be established. . . . . 166

5.7 Errors in matching. Examples of fingerprint images from the same finger that were not correctly matched by our algorithm. (a) and (b) do not match because of the failure of reference point location, (c) and (d) do not match because of the change in inter ridge distances due to finger pressure difference. . . . . 167
5.8 Genuine and imposter distributions for the proposed verification scheme. (a) MSU_DBI database, (b) NIST-9 (Vol. 1, CD No. 1). . . . . 169
5.9 Receiver Operating Characteristic (ROC) curves for two different (filterbank-based and minutiae-based) matchers. (a) MSU_DBI database, (b) NIST-9 (Vol. 1, CD No. 1). FAR and FRR are equal at all points on the Equal-Error Line. Thus, the point of crossing of ROC with this line denotes the equal error rate on the ROC. . . . . 170

6.1 Various Multi-modal Biometric Systems [158]. . . . . 178
6.2 Performance of individual fingerprint matchers. The ROC curves have been averaged over ten runs. . . . . 188
6.3 Normal approximation to the imposter distribution for the matcher *Filter*. (a) Imposter and genuine distributions, (b) ROC curves. Visually, the Normal approximation seems to be good, but causes significant decrease in the performance compared to the nonparametric estimate of the imposter distribution at low FARs. . . . . 189
6.4 Plot of joint scores from matchers *String* and *Filter*. The solid lines denote the three sum rule decision boundaries corresponding to three different thresholds. The dotted lines denote the three product rule decision boundaries corresponding to three different thresholds. . . . 191
6.5 Two-dimensional density estimates for the genuine and imposter classes for $String + Filter$ combination. Genuine density was estimated using Parzen window ($h = 0.01$) estimator and the imposter density was estimated using normalized histograms. . . . . 193
6.6 *ROC* curves for all possible two-matchers combinations. . . . . 194
6.7 Comparison of the proposed combination scheme with the sum and the product rules for the $String + Filter$ combination. . . . . 195
6.8 The performance of the best individual matcher *Dynamic* is compared with various combinations. The $String+Filter$ is the best two-matcher combination and $String + Dynamic + Filter$ is the best overall combination. Note that addition of the matcher *Hough* to the combination $String + Filter$ results in a degradation of the performance. . . . . 196
6.9 Matching scores for the best combination involving *String*, *Dynamic*, and *Filter* matchers. Visually, one can see a small overlap between the genuine ($\circ$) and the imposter ($*$) classes. The class separation statistic is 1.97 for the three-dimensional genuine and imposter densities estimated from these scores. . . . . 197
6.10 Proposed architecture of multi-modal biometrics system based on several fingerprint matchers. . . . . 199

6.11 Performance of matcher combination. (a) & (b) and (c) & (d) were misclassified by the three individual matchers *String*, *Dynamic*, and *Filter* as impostors, but correctly classified as genuine by the combination. Both the minutiae-based and filterbank-based matchers can not deal with large nonlinear deformations, however, a combination of matchers can overcome this. . . . . 200
6.12 Performance improvement by using multiple impressions and multiple fingers. (a) Combining two impressions of the same finger, and (b) combining two fingers of the same person. . . . . 202

7.1 Sample images from the GT database with varying quality index (QI). 0 false minutiae were detected in (a), 7 in (b), and 27 in (c) by the automatic minutiae detection algorithm [11]. . . . . 205
7.2 Examples of images in the GT database. The ground truth minutiae provided by an expert are marked on the image. . . . . 209
7.3 Examples of gray level profiles in the neighborhood of (a) minutiae and (b) non-minutiae. These $32 \times 32$ subimages, scaled to 8 gray levels, are used for training an LVQ. . . . . 211
7.4 Minutiae detection and classification; (a) Minutiae detection using the algorithm in [11] without pruning, (b) results of minutia-pruning; minutiae marked in white were pruned, (c) result of minutia verification instead of pruning; minutiae marked in white were rejected, (d) result of classifying minutiae shown in (b); minutia bifurcations are marked in black and endings are marked in white. . . . . 212
7.5 ROC for fingerprint matching when minutia verification is used. . . . . 213
7.6 ROC for fingerprint matching when minutia classification is used. . . . . 214
7.7 ROC for fingerprint verification when both minutia classification and verification are used. . . . . 215

8.1 The best performance achieved on the MSU_DBI database. The minutiae extraction algorithm of Jain et al. [11] was modified by replacing its post processing stage with minutiae verification stage as described in Chapter 7. Three different matchers, namely, *String*, *Dynamic*, and *Filter*, two different fingers, and three different impressions for each finger of a person were combined. The genuine distribution was estimated using $2,640$ matchings and the imposter distribution was estimated using $95,920$ matchings. Note that the improvement in performance by combining multiple fingers is higher than combining multiple matchers or multiple templates (impressions). This is because different fingers provide the most "independent" information. A simple "sum rule" was used for the combination. . . . . 218

# Chapter 1

# Introduction

## 1.1 Automatic Identification

With the advent of electronic banking, e-commerce, and smartcards and an increased emphasis on the privacy and security of information stored in various databases, *automatic* personal identification has become a very important topic. Accurate automatic personal identification is now needed in a wide range of civilian applications involving the use of passports, cellular telephones, automatic teller machines, and driver licenses. Traditional knowledge-based (password or Personal Identification Number (PIN)) and token-based (passport, driver license, and ID card) identifications are prone to fraud because PINs may be forgotten or guessed by an imposter and the tokens may be lost or stolen. Therefore, traditional knowledge-based and token-based approaches are unable to satisfy the security requirements of our electronically interconnected information society (see Figure 1.1). As an example, a large part of the annual $450 million Mastercard credit card fraud [14] is due to identity fraud. A

perfect identity authentication system will necessarily have a biometric component. Eventually, a foolproof identity authentication systems will have all the three components (knowledge-based, token-based, and biometrics). In this thesis, we have only focused on the biometrics component of an automatic identification system in general, and a fingerprint-based biometric identification system in particular.

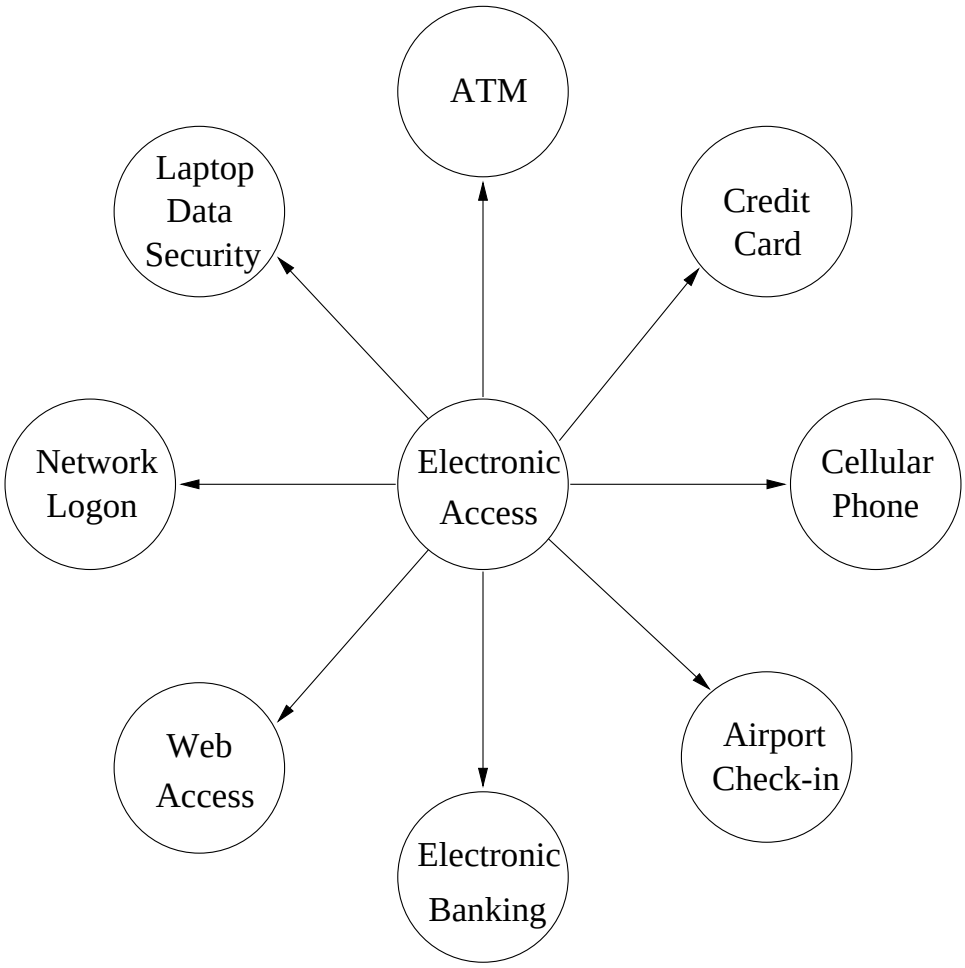



Figure 1.1: Various electronic access applications in widespread use that require automatic authentication.

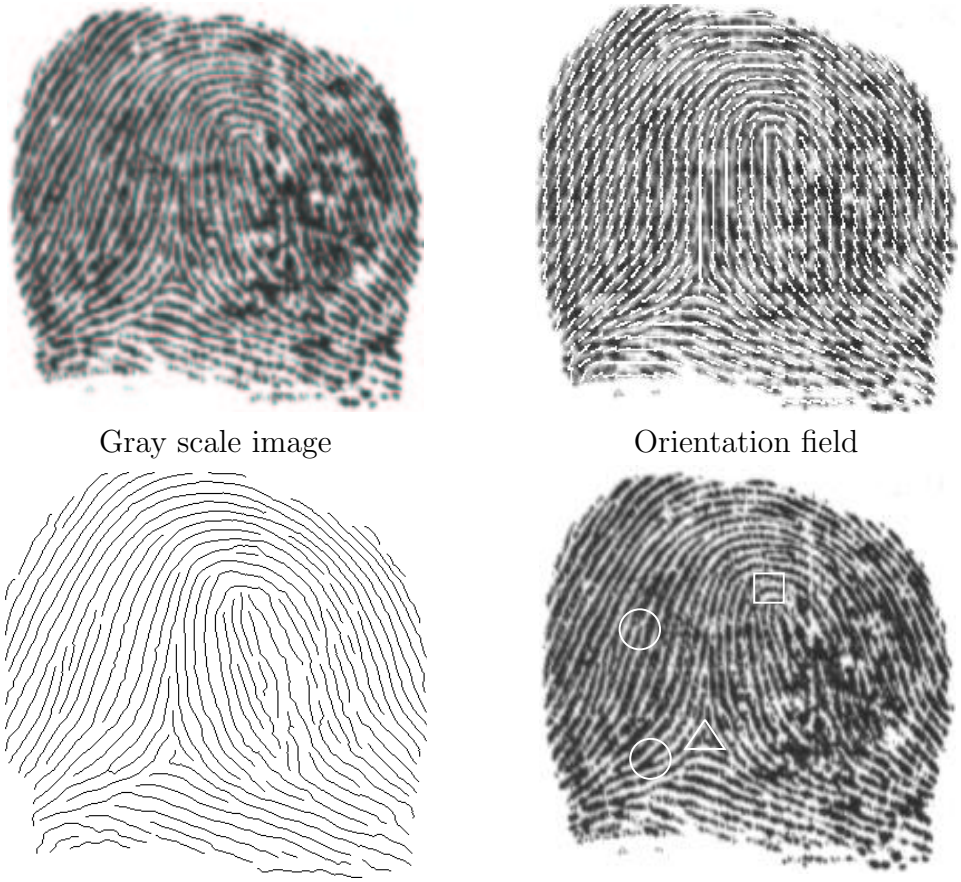



Figure 1.2: Orientation field, thinned ridges, minutiae, and singular points.

## 1.2 Biometrics

*Biometrics*, which refers to identifying an individual based on his or her physiological or behavioral characteristics has the capability to reliably distinguish between an authorized person and an imposter. Since biometric characteristics are distinctive, can not be forgotten or lost, and the person to be authenticated needs to be physically present at the point of identification, biometrics is inherently more reliable and more capable than traditional knowledge-based and token-based techniques. Biometrics also has a number of disadvantages. For example, if a password or an ID card is

compromised, it can be easily replaced. However, once a biometrics is compromised, it is not possible to replace it. Similarly, users can have a different password for each account, thus if the password for one account is compromised, the other accounts are still safe. However, if a biometrics is compromised, all biometrics-based accounts can be broken-in. Among all biometrics (e.g., face, fingerprint, hand geometry, iris, retina, signature, voice print, facial thermogram, hand vein, gait, ear, odor, keystroke dynamics, etc. [14]), fingerprint-based identification is one of the most mature and proven technique.

## 1.3 Applications

Biometrics has been widely used in forensics applications such as criminal identification and prison security. The biometric technology is rapidly evolving and has a very strong potential to be widely adopted in civilian applications such as electronic banking, e-commerce, and access control. Due to a rapid increase in the number and use of electronic transactions, electronic banking and electronic commerce are becoming one of the most important emerging applications of biometrics. These applications include credit card and smart card security, ATM security, check cashing and fund transfers, online transactions and web access. The physical access control applications have traditionally used token-based authentication. With the progress in biometric technology, these applications will increasingly use biometrics for authentication. Remote login and data access applications have traditionally used knowledge-based authentication. These applications have already started using biometrics for person

authentication. The use of biometrics will become more widespread in coming years as the technology matures and becomes more trust worthy. Other biometric applications include welfare disbursement, immigration checkpoints, national ID, voter and driver registration, and time and attendance.

## 1.4 Fingerprints

Fingerprints are the ridge and furrow patterns on the tip of the finger [78] and have been used extensively for personal identification of people [11]. Figure 1.2 shows an example of a fingerprint. The biological properties of fingerprint formation are well understood and fingerprints have been used for identification purposes for centuries. Since the beginning of the 20th century, fingerprints have been extensively used for identification of criminals by the various forensic departments around the world [68]. Due to its criminal connotations, some people feel uncomfortable in providing their fingerprints for identification in civilian applications. However, since fingerprint-based biometric systems offer positive identification with a very high degree of confidence, and compact solid state fingerprint sensors can be embedded in various systems (e.g., cellular phones), fingerprint-based authentication is becoming more and more popular in a number of civilian and commercial applications such as, welfare disbursement, cellular phone access, and laptop computer log-in. The availability of cheap and compact solid state scanners [177] as well as robust fingerprint matchers are two important factors in the popularity of fingerprint-based identification systems. Fingerprints also have a number of disadvantages as compared to other biometrics. For example, ap-

proximately 4% of the population does not have good quality fingerprints, manual workers get regular scratches on their fingers which poses a difficulty to the matching system, finger skin peels off due to weather, fingers develop natural permanent creases, temporary creases are formed when the hands are immersed in water for a long time, and dirty fingers can not be properly imaged with the existing fingerprint sensors. Further, since fingerprints can not be captured without the user's knowledge, they are not suited for certain applications such as surveillance.

## 1.5 Fingerprint Formation

Fingerprints are fully formed at about seven months of fetus development and finger ridge configurations do not change throughout the life of an individual except due to accidents such as bruises and cuts on the finger tips [63]. This property makes fingerprints a very attractive biometric identifier. Biological organisms, in general, are the consequence of the interaction of genes and environment. It is assumed that the phenotype is uniquely determined by the interaction of a specific genotype and a specific environment. Physical appearance and fingerprints are, in general, a part of an individual's phenotype. In the case of fingerprints, the genes determine the general characteristics of the pattern. Fingerprint formation is similar to the growth of capillaries and blood vessels in angiogenesis [63]. The general characteristics of the fingerprint emerge as the skin on the fingertip begins to differentiate. However, the flow of amniotic fluids around the fetus and its position in the uterus change during the differentiation process. Thus, the cells on the fingertip grow in a microenvironment

that is slightly different from hand to hand and finger to finger. The finer details of the fingerprints are determined by this changing microenvironment. A small difference in microenvironment is amplified by the differentiation process of the cells. There are so many variations during the formation of fingerprints that it would be virtually impossible for two fingerprints to be alike. But since the fingerprints are differentiated from the same genes, they will not be totally random patterns either. We could say that the fingerprint formation process is a chaotic system rather than a random one [63].

## 1.6 Fingerprint Individuality

Until recently, the testimony of latent fingerprint examiners was admitted in courts without much scrutiny and challenge. However, in the 1993 case of Daubert vs. Merrell Dow Pharmaceuticals, Inc. [50], the Supreme Court ruled that the reliability of an expert scientific testimony must be established. Additionally, the court stated that when assessing reliability, the following five factors should be considered: $(i)$ whether the particular technique or methodology in question has been subject to a statistical evaluation (hypothesis testing), $(ii)$ whether its error rate has been established, $(iii)$ whether the standards controlling the technique's operations exist and have been maintained, $(iv)$ whether it has been peer reviewed, and published, and $(v)$ whether it has a general widespread acceptance. Subsequently, handwriting identification was challenged under *Daubert* (it was claimed that handwriting identification does not meet the scientific evidence criteria established in the Daubert case) in sev-

eral cases between 1995 and 2001 and many courts have now decided that handwriting identification does not meet the *Daubert* criteria. Fingerprint identification was first challenged by the defense lawyers under *Daubert* in the 1999 case of USA vs. Byron Mitchell [175] on the basis that the fundamental premises of fingerprint identification have not been objectively tested and its potential error rate is not known. The defense motion to exclude fingerprint evidence and testimony was denied. The outcome of the USA vs. Byron Mitchell case is still pending. Fingerprint identification has been challenged under *Daubert* in more than 10 court cases till date since the USA vs. Byron Mitchell case in 1999 (http://onin.com/fp/daubert_links.html).

The two fundamental premises on which fingerprint identification is based are: (*i*) fingerprint details are permanent, and (*ii*) fingerprints of an individual are unique. The validity of the first premise has been established by empirical observations as well as based on the anatomy and morphogenesis of friction ridge skin. It is the second premise which is being challenged in recent court cases. The notion of fingerprint individuality has been widely accepted based on a manual inspection (by experts) of millions of fingerprints. However, the underlying scientific basis of fingerprint individuality has not been rigorously studied or tested. In March 2000, the U.S. Department of Justice admitted that no such testing has been done and acknowledged the need for such a study [174]. In response to this, the National Institute of Justice issued a formal solicitation for "Forensic Friction Ridge (Fingerprint) Examination Validation Studies" whose goal is to conduct "basic research to determine the scientific validity of individuality in friction ridge examination based on measurement of features, quantification, and statistical analysis" [174]. The two main topics of basic research under

this solicitation include: (*i*) measure the amount of detail in a single fingerprint that is available for comparison, and (*ii*) measure the amount of detail in correspondence between two fingerprints.

What do we mean by fingerprint individuality? The fingerprint individuality problem can be formulated in many different ways depending on which one of the following aspects of the problem is under examination: (*i*) the individuality problem may be cast as determining the probability that any two individuals may have sufficiently similar fingerprints in a given target population. (*ii*) Given a sample fingerprint, determine the probability of finding a sufficiently similar fingerprint in a target population. In this thesis, we define the individuality problem as the probability of a false association: given two fingerprints from two different fingers, determine the probability that they are "sufficiently" similar. If two fingerprints originating from two different fingers are examined at a very high level of detail, we may find that the fingerprints are indeed different. However, most human experts and automatic fingerprint identification systems (AFIS) declare that the fingerprints originate from the same source if they are "sufficiently" similar. How much similarity is enough depends on typical (intra-class) variations observed in the multiple impressions of a finger. Solutions to the other two problem formulations (*i*) and (*ii*) above can be derived from a solution to the problem considered in this thesis.

The distinctiveness of fingerprints can be studied by observing the fingerprints of genetically related individuals. The closest genetic relationship is found in monozygotic (identical) twins, and therefore, the maximum similarity between fingerprints is expected to be found among them. A study of identical twin fingerprints can es-

tablish performance bounds on the automatic fingerprint verification systems. In this thesis, we have discussed the implications of the similarity found in identical twin fingerprints on the performance of automatic fingerprint verification systems.

## 1.7 Fingerprint Sensors

The fingerprint images may be acquired either by an offline or an online process. The fingerprint images acquired by the offline process are known as the "inked" fingerprints while the images acquired by the online process are known as "live-scan" fingerprints. Inked fingerprints are of three types: (*i*) rolled, (*ii*) dab, and (*ii*) latent. In the rolled method of fingerprint acquisition, ink is applied to the finger and then rolled on a paper from one side of the nail to the other to form an impression. This paper is then scanned at 500 *dpi* resolution by a standard grayscale scanner. The rolled fingerprints have a larger ridge and furrow area due to the rolling process but have larger deformations due to the inherent nature of the rolling process. In the dab method of fingerprint acquisition, ink is applied to the finger and then pressed onto a paper without rolling. The paper is then scanned into a digital image. Typically, dab inked fingerprints have less nonlinear deformation but smaller area than the rolled inked fingerprints. Latent fingerprints are formed when the fingers leave a thin layer of sweat and grease on the surfaces that they touch due to the presence of sweat pores in our fingertips. Forensic scientists dye this impression which is typically found at the scene of a crime with color and then scan the fingerprint. In this thesis, we have concentrated only on civil applications of fingerprints and therefore, have not used

the latent fingerprints.

A live-scan fingerprint is obtained directly from the finger without the intermediate use of paper (at a resolution of 500 *dpi*). Typically, live-scan sensors capture a series of dab fingerprints when a finger is pressed on the sensor surface. For rolled live-scan fingerprints, the user rolls her/his finger from one end of the nail to the other on the sensor surface and the sensor captures a number of dab fingerprint images. The rolled fingerprint image is then constructed by mosaicking the multiple dab images captured during the rolling process. The commercially available live-scan sensors are based on several different technologies. The optical fingerprint sensor from Digital Biometrics Inc. [54] (model FC21RS1) is based on the "optical total internal reflection" technology. The Thompson-CFS chip-based sensor [163] works on thermal sensing of temperature difference across the ridges and valleys. The Veridicom [177] and the Siemens [156] sensors are based on differential capacitance. The pressure-based and ultrasonic-based fingerprint sensors are available in the market, but they are not very widely used yet.

A number of commercial systems exist that use fingerprints captured by different methods. For example, FBI captures fingerprints of known criminals using the inked rolled method and stores the digitized fingerprint images in its database. A suspect's latent fingerprint found at a scene of crime is then matched to the rolled inked fingerprints in the database. As another example, MasterCard instructs the new credit card applicants to make an inked rolled impression of their finger on a paper and mail the paper to them. The inked rolled fingerprint is then scanned and stored in the user's credit card. The user is then verified at the time of credit card transactions

(a)

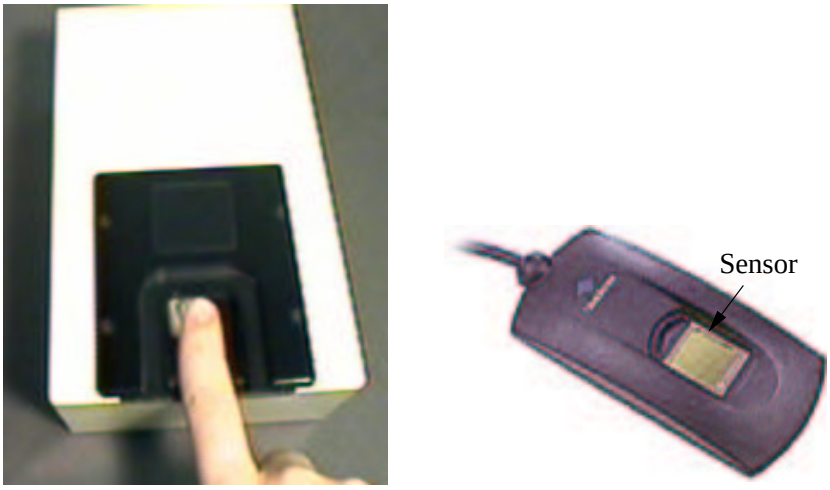
(b)

(c)

Figure 1.3: Fingerprint images captured using (a) inked method (NIST-9 database), image size = 832 × 768 pixels, (b) Digital Biometrics optical sensor (MSU_DBI database), image size = 508 × 480 pixels, and (c) Veridicom solid-state sensor (MSU_VERIDICOM database), image size = 300 × 300 pixels. All the images have 256 gray levels.

using a dab live-scan fingerprint image obtained with the live-scan fingerprint scanner attached to the ATM.

An additional point worth mentioning in this section is that the FBI has prescribed a standard resolution of 500!*dpi* for fingerprint images. A large number of live fingerprint sensors available in the market today operate at this resolution. National Institute of Standards and Technology (NIST) provides a number of fingerprint databases to the research community for benchmark purposes. A number of these databases contain inked rolled fingerprints (e.g., NIST-4, NIST-9, etc). These databases contain fingerprint images scanned at 500 *dpi* from the paper copy of the rolled impressions as well as captured by 500 *dpi* live scanners. A few sensors that image the fingerprints at a lower resolution are also available in the market. However, since 500 *dpi* resolution is the standard, we use fingerprint images scanned only at this resolution in this thesis.




(a) (b)

Figure 1.4: Fingerprint sensors. (a) Optical sensor from Digital Biometrics, Inc., and (b) solid-state sensor from Veridicom, Inc.

Figure 1.3(a) shows a fingerprint image captured using the inked method. The NIST 9 database, CD. No. 1, contains 900 fingerprint images captured by this method. Figures 1.3(b) and (c) show fingerprint images captured by the optical live-scan sensor manufactured by Digital Biometrics, Inc. (Figure 1.4(a)) and solid-state live-scan fingerprint sensor manufactured by Veridicom, Inc. (Figure 1.4(b)). The inked method captures the largest fingerprint area. The chip-based sensors capture only a part of the whole fingerprint due to their small size. Two images of the same finger may capture different parts of the fingerprint. Due to this relatively small overlap between different images of the same finger captured with the small sensors, the fingerprint matching problem is challenging. However, due to their small size (see Figure 1.4), the solid-state sensors can be easily embedded into laptops, cellular phones, mouse and firearms.

## 1.8 Fingerprint Representation

The popular fingerprint representation schemes have evolved from an intuitive system developed by forensic experts who visually match the fingerprints. These schemes are either based on predominantly local landmarks (e.g., minutiae-based fingerprint matching systems [11, 56]) or exclusively global information (fingerprint classification based on the Henry system [18, 76, 105]). The minutiae-based automatic identification techniques first locate the minutiae points and then match their relative placement in a given finger and the stored template [11]. A good quality inked fingerprint image contains between 60 to 80 minutiae, but different fingerprints and different

acquisitions of the same finger have different numbers of minutiae. A graph-based representation [118, 155, 5] constructs a nearest neighbor graph from the minutiae patterns. The matching algorithm is based on inexact graph matching techniques. The point pattern-based representation [11, 26, 96] considers the minutiae points as a two-dimensional pattern of points. Correlation-based techniques [61, 31] consider the gray level information in the fingerprint as features and match the global patterns of ridges and valleys to determine if the ridges align.

The global representation of fingerprints (e.g., whorl, left loop, right loop, arch, and tented arch) is typically used for indexing [18, 76, 105], and does not offer good individual discrimination. Further, the indexing efficacy of existing global representations is poor due to a small number of categories (typically five) that can be effectively identified automatically and a highly skewed distribution of the population in each category. The global representation schemes of the fingerprint used for classification can be broadly categorized into four main categories: (*i*) knowledge-based, (*ii*) structure-based, (*iii*) frequency-based, and (*iv*) syntactic. The knowledge-based fingerprint representation technique uses the locations of singular points (core and delta) to classify a fingerprint into five major classes (whorl, left loop, right loop, arch, and tented arch) [18, 105]. A knowledge-based approach tries to capture the knowledge of a human expert by deriving rules for each category by hand-constructing the models and therefore, does not require training. Structure-based approach uses the estimated orientation field in a fingerprint image [30, 122]. Frequency-based approaches use the frequency spectrum of the fingerprints for representation [25]. Hybrid approaches combine two or more approaches for representation [34, 120].

There are two major shortcomings of the traditional approaches to fingerprint representation. For a significant fraction of the population, the automatic extraction of representations based on an explicit detection of complete ridge structures in the fingerprint is difficult. The widely used minutiae-based representation does not utilize a significant component of the rich discriminatory information available in the fingerprints. Local ridge structures cannot be completely characterized by minutiae. Further, minutiae-based matching has difficulty in efficiently and robustly matching two fingerprint images containing different numbers of unregistered minutiae points. Some applications such as smart cards will also benefit from a compact representation.

## 1.9 Fingerprint Classification

Large volumes of fingerprints are collected and stored everyday in a wide range of applications, including forensics, access control, and driver license registration. Automatic identity recognition based on fingerprints requires that the input fingerprint be matched with a large number of fingerprints stored in a database (the FBI database currently contains more than 630 million fingerprints! [69]). To reduce the search time and computational complexity, it is desirable to classify these fingerprints in an accurate and consistent manner such that the input fingerprint needs to be matched only with a subset of the fingerprints in the database. Fingerprint classification is a technique used to assign a fingerprint into one of the several pre-specified types already established in the literature (and used in forensic applications) which can provide an indexing mechanism. Fingerprint classification can be viewed as a coarse

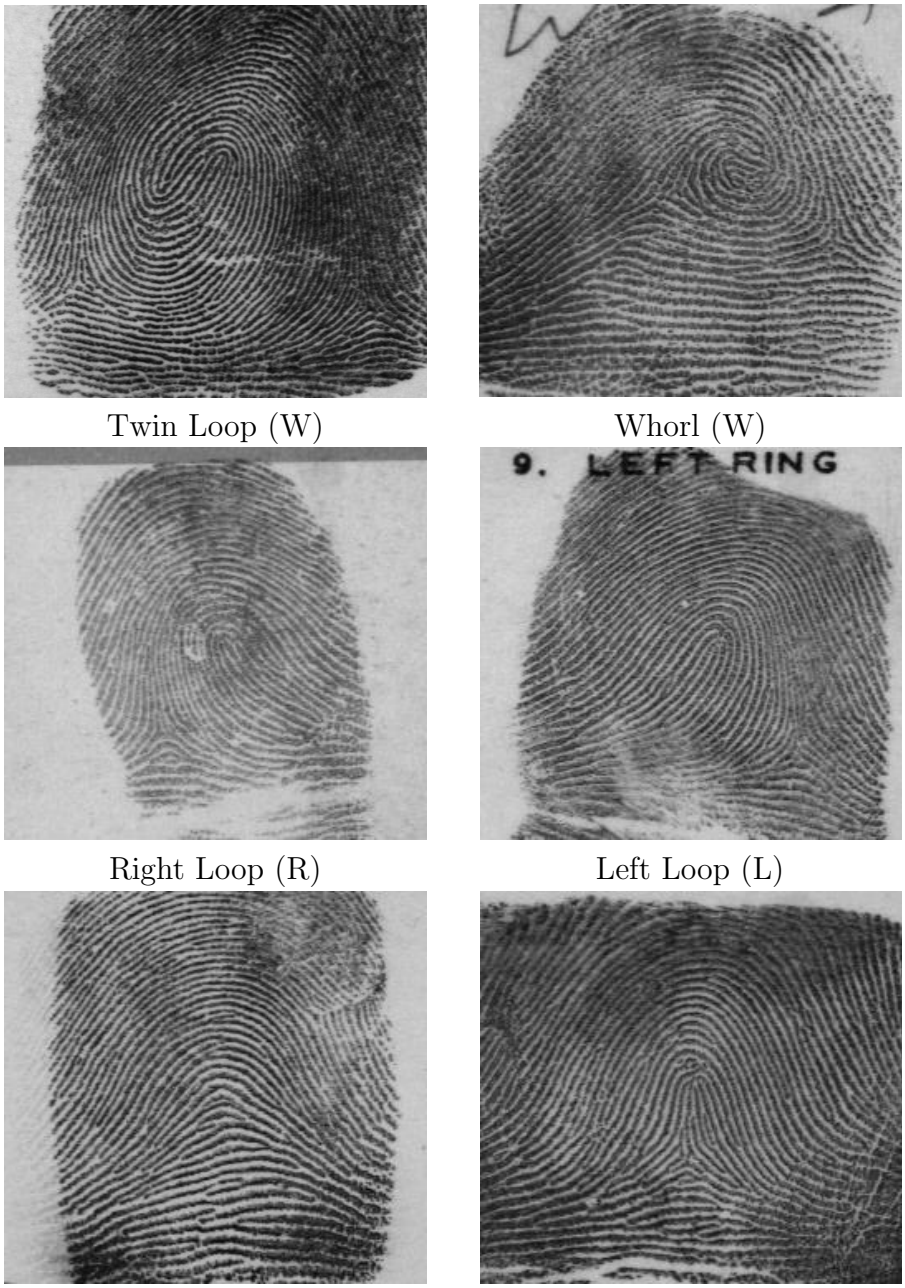



Figure 1.5: Six major fingerprint classes. Twin loop images are labeled as whorl in the NIST-4 database.

level matching of the fingerprints. An input fingerprint is first matched to one of the pre-specified types and then it is compared to a subset of the database corresponding to that fingerprint type. To increase the search efficiency, the fingerprint classification algorithm can classify a fingerprint into more than one class. For example, if the fingerprint database is binned into five classes, and a fingerprint classifier outputs two classes (primary and secondary) with high accuracy, then the identification system will only need to search two of the five bins, thus decreasing the search space 2.5 folds. Continuous classification of fingerprints is also very attractive for indexing where fingerprints are not partitioned in non-overlapping classes, but each fingerprint is characterized with a numerical vector summarizing its main features. The continuous features obtained are used for indexing fingerprints through spatial data structures and for retrieving fingerprints by means of spatial queries [22]. In this thesis, we have concentrated on an exclusive fingerprint classification and classify fingerprints into five distinct classes, namely, *whorl* ($W$), *right loop* ($R$), *left loop* ($L$), *arch* ($A$), and *tented arch* ($T$) (Figure 1.5). The five classes are chosen based on the classes identified by the National Institute of Standards and Technology (NIST) to benchmark automatic fingerprint classification algorithms. The natural proportion of occurrence of these five major classes of fingerprints is 0.3252, 0.3648, 0.1703, 0.0616, and 0.0779 for whorl, right loop, left loop, arch, and tented arch, respectively [173].

There are two main types of features in a fingerprint: ($i$) global ridge and furrow structures which form special patterns in the central region of the fingerprint, and ($ii$) local ridge and furrow minute details (see Figure 1.2). A fingerprint is classified based on only the first type of features and is uniquely identified based on the second type

of features (ridge endings and bifurcations, also known as minutiae). See Figure 1.2 for examples of ridges, minutiae, orientation field and singular points in a fingerprint image.

## 1.10 Fingerprint Verification

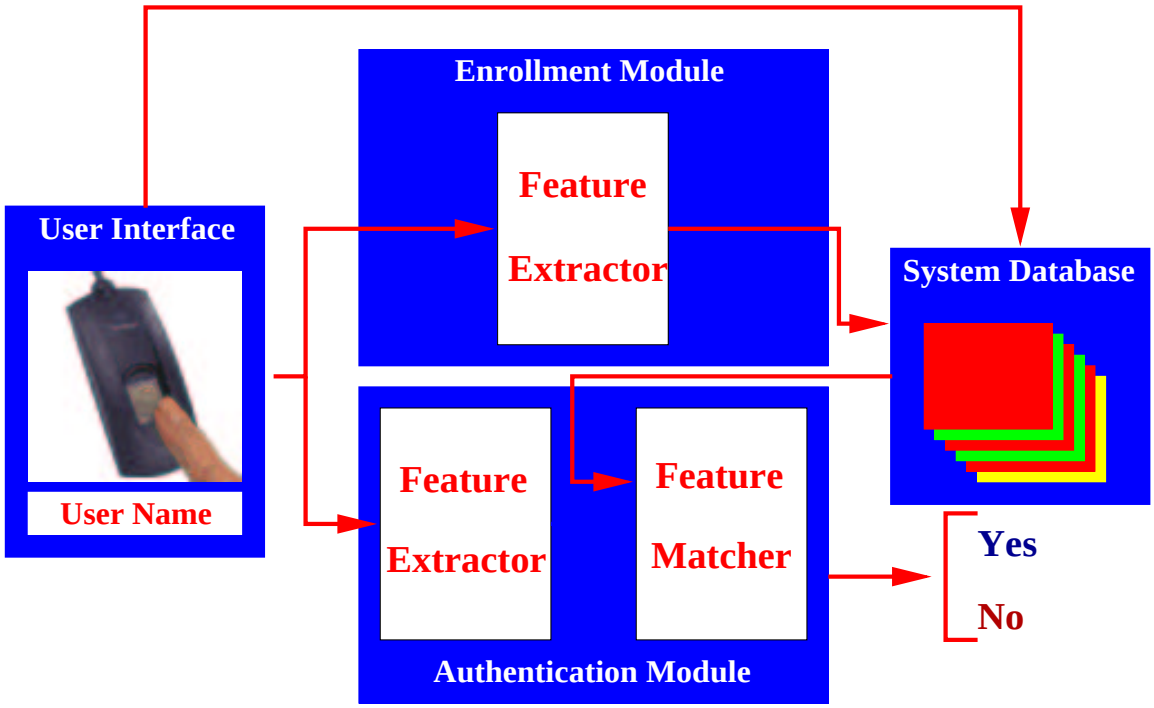



Figure 1.6: System diagram for an automatic verification system.

A biometric system can be operated in two modes: 1) verification mode and 2) identification mode. In the verification mode, a biometric system either accepts or rejects a user's claimed identity while a biometric system operating in the identification mode establishes the identity of the user without a claimed identity. Fingerprint identification is a more difficult problem than fingerprint verification because a huge number of comparisons needs to be performed in identification. In this thesis, we

have focused on a biometric system operating in a verification mode and an indexing scheme (fingerprint classification) that can be used in an identification system. A number of civilian applications operate in verification mode on a regular basis and perform identification only at the time of the user registration to check the integrity of the database (e.g., finding duplicates). For example, in an ATM application, after a user has been registered and issued an ATM card, the acquired fingerprint needs to be matched only with a single template fingerprint stored on the ATM card on each transaction. A typical verification system can be divided into two modules: (*i*) enrollment and (*ii*) verification. The enrollment module scans the fingerprint of a person through a sensing device and then stores a representation (called template) of the fingerprint in the database. The verification module is invoked during the operation phase. The same representation which was used in enrollment phase is extracted from the input fingerprint and matched against the template of the claimed identity to give a "yes/no" answer. On the other hand, an identification system matches the input fingerprint with a large number of fingerprints in the database and as a result, fingerprint classification is effective only in an identification system and is not an issue in a verification system. In this thesis, we have used the term "identification" in a loose sense for both the fingerprint verification and identification problems and the exact meaning of the term can be resolved based on the context.

The biometric verification problem can be formulated as follows. Let the stored biometric signal (template) of a person be represented as $S$ and the acquired signal (input) for authentication be represented by $I$. Then the null and alternate hypotheses

can be stated as:

$$H_0: \ I \neq S, \ \text{input fingerprint is NOT the same as the template,}$$

$$H_1: \ I = S, \ \text{input fingerprint is the same as the template.}$$

The associated decisions are as follows:

$$D_0: \quad \text{person is an imposter,}$$

$$D_1: \quad \text{person is genuine.}$$

The verification involves matching $S$ and $I$ using a similarity measure. If the similarity/matching score is less than some decision threshold $T$, then decide $D_0$, else decide $D_1$. The above terminology is borrowed from communications theory where we want to detect a message in the presence of noise. $H_0$ is the hypothesis that the received signal is noise alone and $H_1$ is the hypothesis that the received signal is message plus the noise. Such a hypothesis testing formulation inherently contains two types of errors: Type I: false acceptance ($D_1$ is decided when $H_0$ is true) and Type II: false rejection ($D_0$ is decided when $H_1$ is true). The two types of errors are also known as FAR and FRR, defined as:

$$\begin{aligned} False\ Accept\ Rate &= P(D_1|w_0), \\ False\ Reject\ Rate &= P(D_0|w_1), \end{aligned}$$

where $w_0$ is the class associated with $H_0 = true$ and $w_1$ is the class associated with $H_1 = true$. The performance of a biometric system is usually specified in terms of its FAR. The decision scheme should establish a decision boundary which minimizes the FRR for the specified FAR. There is a trade-off between the two types of errors and both the errors cannot be reduced simultaneously based on the operating point alone. The given biometric application dictates the FAR and FRR requirements for the verification system. For example, access to an ATM machine generally needs a small FRR, but access to a secure military installation requires a very small FAR.

## 1.11 Information Fusion

A number of fingerprint verification systems have been developed and tested on large databases but most of them are not able to meet the rigid performance requirements in high security applications. Each fingerprint verification system uses different feature extraction and/or matching algorithms to generate a matching score which is used for authentication. It is well known in the pattern recognition literature that different classifiers often misclassify different patterns [164, 90]. This suggests that different classifiers offer rather complementary information about the given classification task. A combination scheme which harnesses various information sources is likely to improve the overall system performance. The outputs of various classifiers can be combined to obtain a decision which is more accurate than the decisions made by any one of the individual classifiers. Similar ideas can be used to combine different fingerprint matching algorithms as described in Chapter 6.

## 1.12 Feature Verification

Ideally, we would like to design pattern recognition systems which make decisions based on *all* the information available in the input image. However, traditionally, for simplicity of design, a sequential approach is often adopted to feature extraction and matching, where each stage transforms a particular component of the information relatively independently and the interaction between these components of information is limited. Often, the rather simplistic model used in each component (stage) is not sufficient to capture the entire sensed data. One of the problems with the sequential approach is that the limited use of information in each stage results in feature extraction and matching performance artifacts. Even though the sequential approach is efficient from design and processing point of view, it may introduce errors in the feature extraction and recognition stages. We believe that by reexamining the original image data, some of the errors in the end-to-end sequential processing can be eliminated, resulting in an improvement in system performance. The main limitation of the feature verification algorithm is that it cannot address the problem of missed features. Therefore, the feature detection algorithm should be operated at a very low false reject rate at the expense of higher false accept rate. The false accepts of the feature extraction algorithm will be verified by the feature verification algorithm. Performance can also be improved by feature refinement. See Figure 1.7 for our proposed modifications to a sequential feature extraction system.

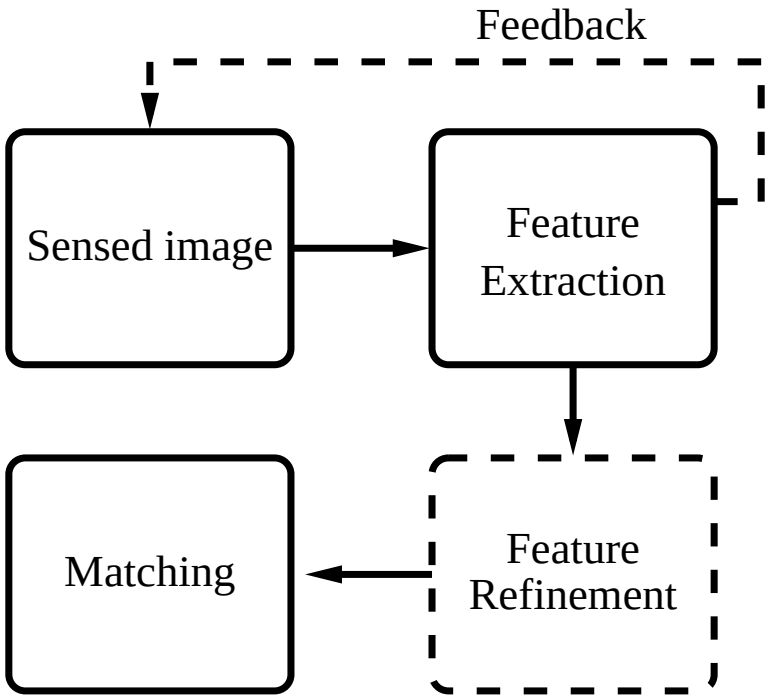



Figure 1.7: A general pattern recognition system with proposed feedback in feature extraction and a new feature refinement stage.

## 1.13 Challenges in Automatic Fingerprint Identification

Even though several commercial systems exist for fingerprint-based identification [177], the matching accuracy performance is still not acceptable in many emerging civilian applications. A fingerprint identification system involves several stages. First, the fingerprint image needs to be acquired and scanned into a digital representation. There is a loss of information when the three-dimensional fingerprint is scanned into a two-dimensional digital image. Placement of the finger on the sensor, cuts and bruises on the finger and finger pressure differences cause different impressions of the fingerprint to appear different. It is a challenge for the feature extraction algorithm to reliably extract a robust representation from these images. Due to the

noise present in the fingerprint image because of inexact sensing process, there may be false features detected or important features missed. The matching algorithm should recover the invariant information from the features such that it outputs a high score when matching impressions of the same finger and a low score when matching the impressions of different fingers. If the fingerprint image is of poor quality, a fingerprint enhancement algorithm should be used to improve the quality of the image. However, it is very difficult to design a fingerprint enhancement algorithm that is robust to all types of noise in the sensed fingerprint. An inappropriate enhancement algorithm may introduce undesirable artifacts into the fingerprint image.

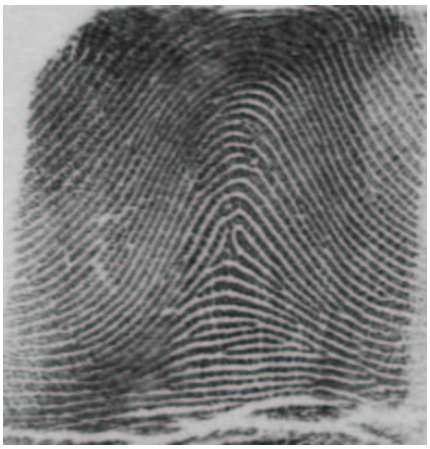


Figure 1.8: An example fingerprint image from the NIST-4 database. The experts have labeled this image to belong to two classes, right loop, and tented arch.

In a verification application, it is very important to make a decision in real time ($\sim$ 1 second) so that the verification process does not cause inconvenience to the user. In an identification application, the fingerprint matching should be extremely fast due to the large number of matchings that must be performed. The matching algorithm should scale well with large databases, both in terms of time and space. Fingerprint classification can be used to distribute the fingerprints in a fixed number of bins so

that the matching algorithm needs to search only a few bins to find the correct match. FBI requirements for a fingerprint classification algorithm are 1% error rate with a maximum of 20% reject rate. Fingerprint classification is a difficult problem for both the automatic systems and the human experts. For example, about 17% of the images in the NIST-4 database [41] have two different ground truth labels. This means that even human experts could not agree on the true class of the fingerprint for about 17% of the fingerprint images in this database containing $4,000$ fingerprint images (see Figure 1.8 for an example).

## 1.14 State-of-the-art in Fingerprint Identification

A number of systems exist for fingerprint verification as well as classification. Even though National Institute of Standards and Technology (NIST) provides a number of databases for performance evaluation and benchmark, many companies report results on their proprietary databases and, therefore, their results cannot be independently verified and compared. Some of the fingerprint vendors report extremely low error rates (see Table 1.1) that are not achieved in research laboratories on realistic databases. As a comparison, a recent evaluation of various fingerprint verification algorithms on a common database in a laboratory environment reports significantly higher error rates (Table 1.2). The details of this performance evaluation can be found in [57].

A state-of-the-art fingerprint classification algorithm [141] reports accuracies of 92.2% for the five-class classification problem with classes defined as arch, tented arch,

Table 1.1: Performance of fingerprint verification systems reported by various companies on their web sites. None of the companies mention the database used for obtaining the performance results, and thus the performance numbers can not be directly compared. FAR: False Accept Rate; FRR: False Reject Rate.

| Company (web site) | Sensor | FAR (%) | FRR (%) |
|---|---|---|---|
| Biolink USA (biolinkusa.com) | Optical | 0.0000001 | 0.01 |
| BiometricId (biometricid.com) | Optical | 0.01 | 0.01 |
| Startek (startek.com.tw) | Optical | 0.001 | 3.3 |
| IOSoftware (iosoftware.com) | Optical | 0.1 | 1 |
| Identix (identix.com) | Optical | 0.0001 | 1 |
| NEC (nectech.com) | Solid-state | 0.0002 | 0.05 |
| Biometrix Int. (biometrix.at) | Solid-state | 0.001 | 0.0001 |
| Pollex (pollex.ch) | Solid-state | 0.001 | 1 |
| Sony (sony.com) | Solid-state | 0.001 | 1 |

left loop, right loop, and whorl, and 94.5% for the four-class classification, where the classes arch and tented arch are merged into one. The state-of-the-art classification systems have not met the FBI standards on any public domain database containing equal number of patterns from each of the five fingerprint classes.

## 1.15 Thesis Objectives

Forensic experts who match fingerprints visually have predominantly used minutiae features for fingerprint matching for over a century. Similarly, forensic experts have used the locations of singularities in the fingerprints (e.g., core(s) and delta(s)) to visually classify fingerprints for indexing purposes. Most of the existing automatic fingerprint verification and classification systems use representations that are motivated by the representations used by the forensic experts. In this thesis, we have

Table 1.2: Comparison of state-of-the-art fingerprint verification algorithms in terms of equal error rate (ERR) and timing on a database of 800 fingerprints (image size = $448 \times 478$ captured by DF-90 optical sensor manufactured by Identicator Technology). Details of the evaluation protocol can be found in [57].

| Algorithm | ERR (%) | Avg Enroll Time (seconds) | Avg Match Time (seconds) | Reject Rate (%) |
|---|---|---|---|---|
| Sag1 | 3.64 | 5.70 | 2.13 | 0.00 |
| Sag2 | 4.01 | 1.94 | 1.94 | 0.00 |
| Cspn | 5.33 | 0.35 | 0.35 | 1.81 |
| Cetp | 8.29 | 1.49 | 1.66 | 0.00 |
| Cwai | 5.90 | 0.46 | 0.57 | 20.86 |
| Krdl | 8.03 | 1.48 | 1.60 | 11.98 |

theoretically determined the information content of the traditional minutiae-based representation and established an upper bound on the performance of fingerprint verification systems based on a minutiae representation. As a result of the limited information content of the minutiae representation, non-minutiae representations of fingerprints should be explored. In this thesis, we have developed a novel non-minutiae representation for fingerprints that combines both the global and the local information present in a fingerprint. The proposed representation is based on considering the fingerprint images as oriented textures, is very different from the representations used by the forensic experts and is more amenable to automatic systems (in terms of matching speed and storage size). The performance of this representation is evaluated for both fingerprint classification and matching applications on large databases. We have empirically shown that the proposed representation has a discriminatory power that is comparable to the minutiae-based representation. A combination of a matcher based on the proposed representation with two other minutiae-based match-

ers significantly improves the verification performance. We have further shown that a combination of multiple templates and multiple fingers can significantly improve the performance of a fingerprint verification system. A feedback and feature refinement scheme is proposed in a general pattern recognition framework which improves the performance of a minutiae-based fingerprint verification system. Finally, we show that the use of all the techniques presented in this thesis significantly improve the performance of a fingerprint verification system on a large database.

## 1.16 Thesis Outline

Chapter 2 discusses the individuality of fingerprints. Chapter 3 describes our novel filterbank-based fingerprint representation. A classification algorithm based on the proposed representation is described in Chapter 4. Chapter 5 describes the filterbank-based fingerprint verification system and compares it with a state-of-the-art minutiae-based system. Chapter 6 presents a classifier combination strategy geared towards decision level fusion in fingerprint verification systems. The results of combining four different fingerprint matchers, two fingers of a person and two impressions of the same fingerprint are presented. Chapter 7 presents results of minutiae verification and classification. Chapter 8 presents conclusions and future directions.

# Chapter 2

# On the Individuality of Fingerprints

Fingerprint identification is based on two basic premises: (i) persistence: the basic characteristics of fingerprints do not change with time; and (ii) individuality: the fingerprint is unique to an individual. The validity of the first premise has been established by the anatomy and morphogenesis of friction ridge skin. While the second premise has been generally accepted to be true based on empirical results, the underlying scientific basis of fingerprint individuality has not been formally tested. As a result, fingerprint evidence is now being challenged in several court cases. A scientific basis for establishing fingerprint individuality will not only determine the admissibility of fingerprint identification in the courts of law but will also establish an upper bound on the performance of an automatic fingerprint verification system.

The distinguishing nature of physical characteristics of a person is due to both the inherent individual genetic diversity within the human population as well as the

random processes affecting the development of the embryo [146, 129]. Since two individuals can be arbitrarily close with respect to their genetic constitution (e.g., identical twins), a pessimistic evaluation of identity discrimination based on biometrics may need to rely solely on an assessment of diversity in the traits due to random process affecting human development. Such an assessment strategy would necessarily rely on biometric samples from individuals who are identical/similar in their genetic constitution. Since identical twins have the closest genetics-based relationship, the maximum similarity between fingerprints is expected to be found among them. In Section 2.1, we have quantified the role of genetic similarity on the similarity of fingerprints and shown that a state-of-the-art automatic fingerprint identification system can successfully distinguish identical twins though with a slightly lower accuracy than nontwins [21]. The implications of the similarity found in identical twin fingerprints on the performance of fingerprint identification systems is discussed.

The environmental factors during the formation of fingerprints play an important role in the distinctiveness of fingerprints. To quantify the diversity present in fingerprint patters, we study the amount of information available in minutiae points to establish a correspondence between two fingerprint images in Section 2.2. We derive an expression which estimates the probability of falsely associating minutiae-based representations from two arbitrary fingerprints. For example, we show that the probability that a fingerprint with 36 minutiae points will share 12 minutiae points with another arbitrarily chosen fingerprint with 36 minutiae points is $6.10 \times 10^{-8}$. These probability estimates are compared with typical fingerprint matcher accuracy results. Our results show that (i) contrary to the popular belief, fingerprint matching is not

infallible and leads to some false associations, (ii) the performance of automatic fingerprint matcher does not even come close to the theoretical performance, and (iii) due to the limited information content of the minutiae-based representation, the automatic system designers should explore the use of non-minutiae-based information present in the fingerprints.

## 2.1 Genetic Factors

### 2.1.1 Introduction

The extent of variation in a physical trait due to random development process differs from trait to trait. By definition, identical twins can not be distinguished based on DNA. Typically, most of the physical characteristics such as body type, voice, and face are very similar for identical twins and automatic identification based on face and hand geometry is unlikely to distinguish them. See Figure 2.1 for a photograph of an identical twin pair. It is, however, claimed that identical twins can be distinguished based on their fingerprints, retina, thermogram, or iris patterns. The focus of this study is to empirically determine the similarity of fingerprints in identical twins. We further attempt to assess the impact of this similarity on the performance of automatic fingerprint-based verification systems. Since both, human iris and angiogenesis follow a development pattern similar to fingerprints, we believe the results of this study may be qualitatively applicable to other biometric identifiers such as iris, retina and thermogram patterns as well.

Figure 2.1: Photograph of identical twin sisters (www.visi.com/∼charlesr/).

How does one assess whether two fingerprints are identical? In order to reliably establish whether two prints came from the same finger or different fingers, it is necessary to capture some *invariant* representation (features) of the fingerprints: the features which over a life-time will continue to remain unaltered irrespective of the cuts and bruises, the orientation of the print with respect to the medium of the capture, occlusion of a small part of the finger, the imaging technology used to acquire the fingerprint from the finger, or the elastic distortion of the finger during the acquisition of the print.

An important question in fingerprint matching is: which characteristics of the fingerprints are inherited? A number of studies have shown a significant correlation in the fingerprint class (i.e., whorl, right loop, left loop, arch, tented arch) of identical twin fingers; correlation based on other generic attributes of the fingerprint such as ridge count, ridge width, ridge separation, and ridge depth has also been found to

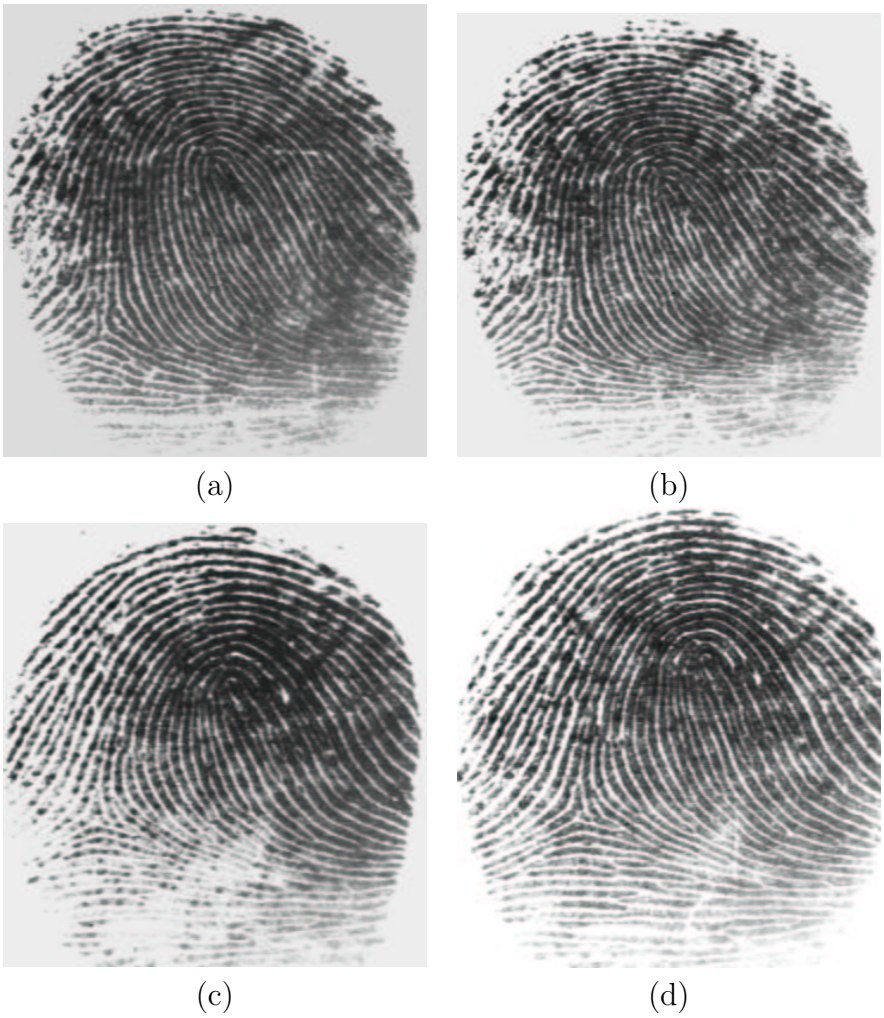


Figure 2.2: Fingerprint images of identical twin sisters captured using an optical scanner from Digital Biometrics Inc., (a) and (b) are two impressions of the same finger of one twin and (c) and (d) are two impressions of the corresponding finger of her sibling. Matching score between (a) and (b) is 487, and between (c) and (d) is 510. The matching score between (a) and (c) is 24, and the matching score between (b) and (d) is 4. The fingerprints of both the twins here have the same type (right loop) and look similar to untrained eyes. Fingerprint experts, as well as our automatic fingerprint identification system can, however, easily differentiate the twins.

be significant in identical twins. In dermatoglyphics studies, the maximum global difference between fingerprints has been found among individuals of different races. Unrelated persons of the same race have very little global similarity in their fingerprints, parent and child have some global similarity as they share half the genes, siblings have more similarity and the maximum global similarity is observed in the monozygotic (identical) twins, which is the closest genetic relationship [79].

Monozygotic twins are a consequence of division of a single fertilized egg into two embryos. Thus, they have exactly identical DNA except for the generally undetectable micromutations that begin as soon as the cell starts dividing. Fingerprints of identical twins start their development from the same DNA, so they show considerable generic similarity [178]. However, identical twins are situated in different parts of the womb during development, so each fetus encounters slightly different intrauterine forces from their siblings. As a result, fingerprints of identical twins have different microdetails which can be used for identification purposes [79]. It is claimed that a trained expert can usually differentiate between the fingerprints of identical twins based on the minutiae (dis)similarity [79]. Thus, there is anecdotal evidence that minutiae configurations are different in identical twins but to our knowledge, no one has systematically investigated or quantified how minutiae information in identical twins is (un)related in the context of an automatic fingerprint-based authentication system. The multiple fingerprints of a single individual also share common genetic information and a very common development environment. However, this chapter focuses on analyzing the similarity in fingerprint minutiae patterns in identical twin fingers.

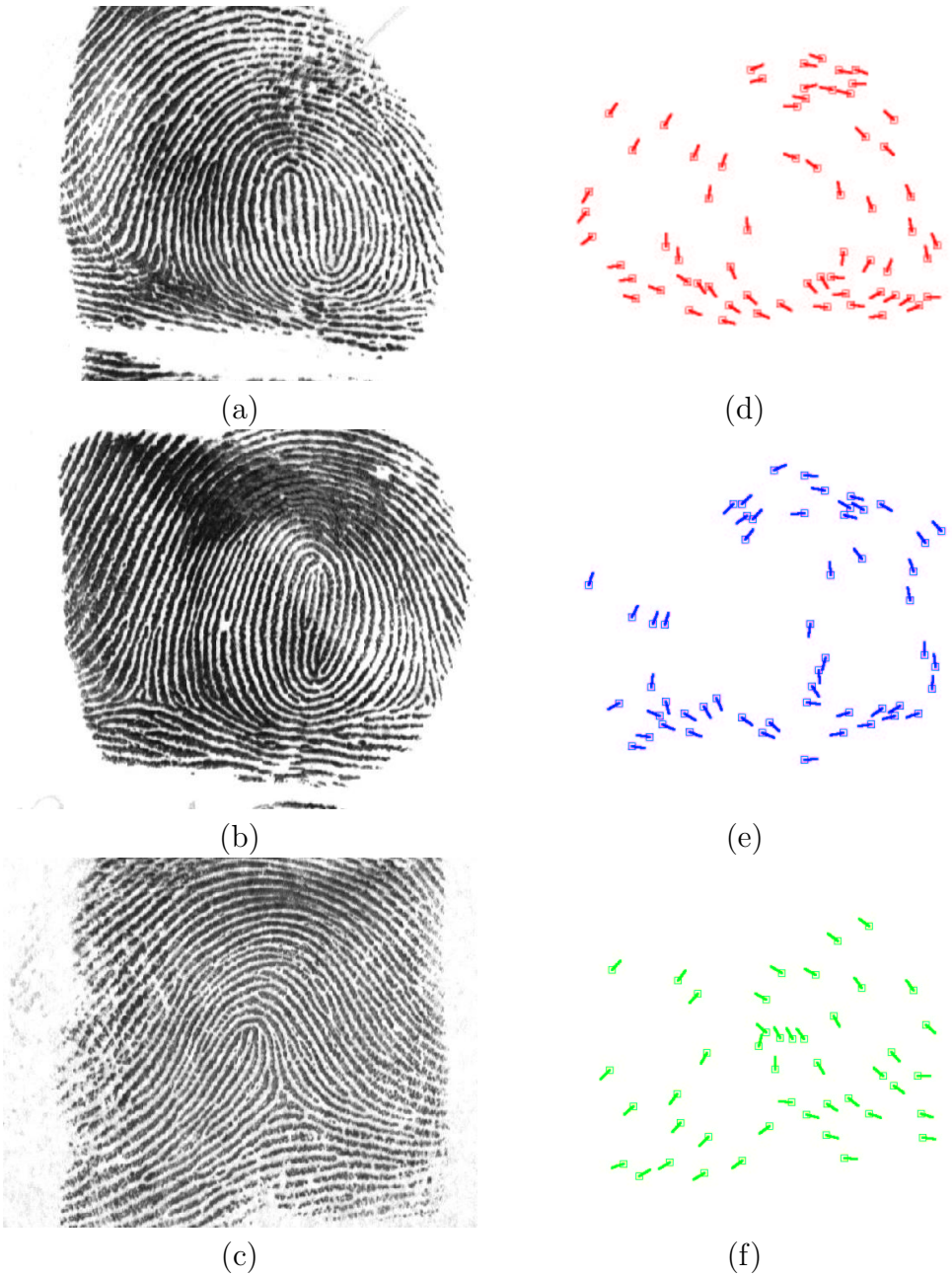
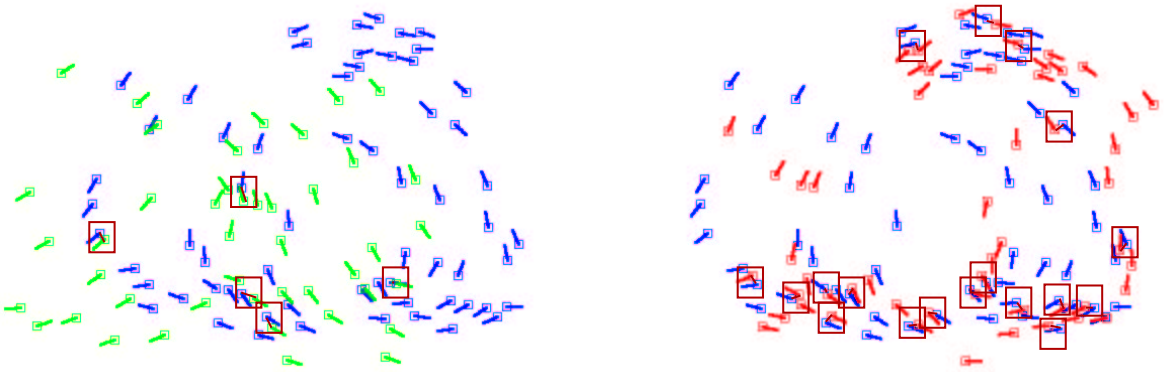



Figure 2.3: Minutiae extraction for twins. (a) and (b) are fingerprint images of an identical twin and his/her sibling while the fingerprint in (c) is from another person. (d), (e), and (f) are the minutiae extracted from (a), (b), and (c), respectively using the extraction algorithm in [11].



(a) (b)

Figure 2.4: Minutiae matching for (a) twin-nontwin (matching of Figures 2.3(e) and 2.3(f), matching score = 3 on a scale of 0-999) and (b) twin-twin (matching of Figures 2.3(d) and Figure 2.3(e), matching score = 38 on a scale of 0-999). The "matched" minutiae pairs are shown by bounding boxes.

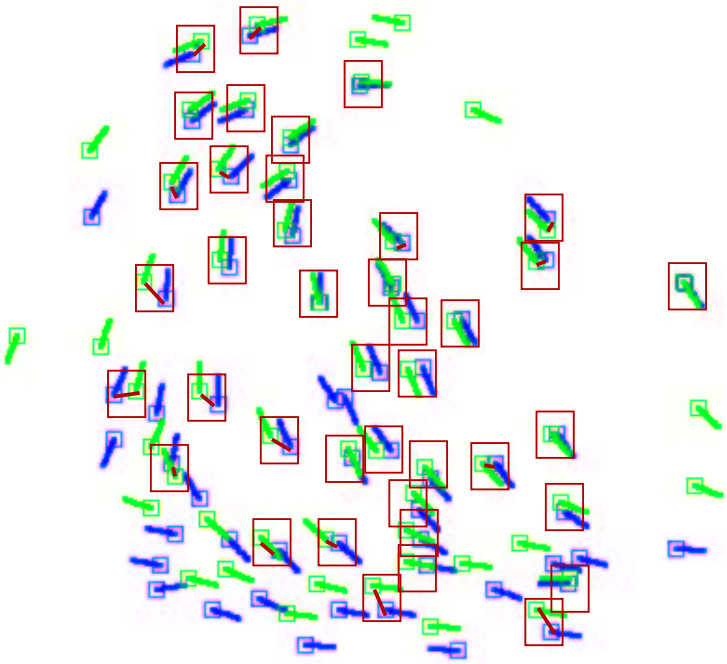


Figure 2.5: Minutiae matching for two impressions of the same finger shown in Figures 2.2(a) and 2.2(b) (matching score = 487 on a scale of 0-999). The "matched" minutiae pairs are shown by bounding boxes.

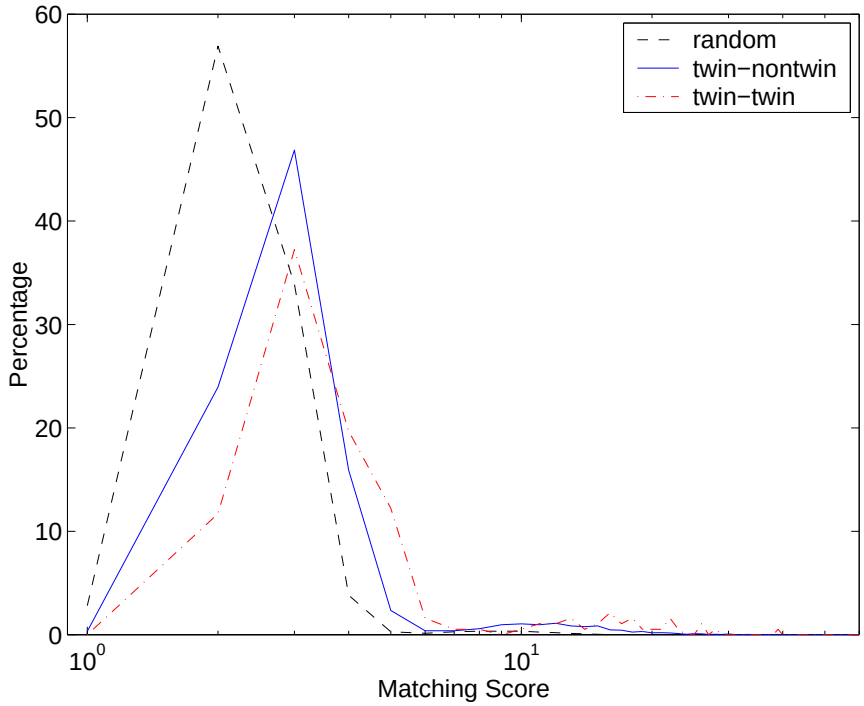
Using an automatic fingerprint biometric system [11], we study the (dis)similarity between identical twin fingerprints and compare it to the (dis)similarity between two arbitrary fingerprints. We have confirmed the claim that the identical twin fingerprints have a large class correlation, i.e., if one of the identical twin's fingerprint is a whorl then it is very likely that the other twin's fingerprint will also be of whorl type. We also analyze the correlation between the fingerprint class and the minutiae matching score between two randomly chosen fingerprints. Finally, we stipulate the implications of the extent of the similarity in identical twin fingerprints to the performance of a fingerprint-based person verification system.

### 2.1.2 Experimental Results

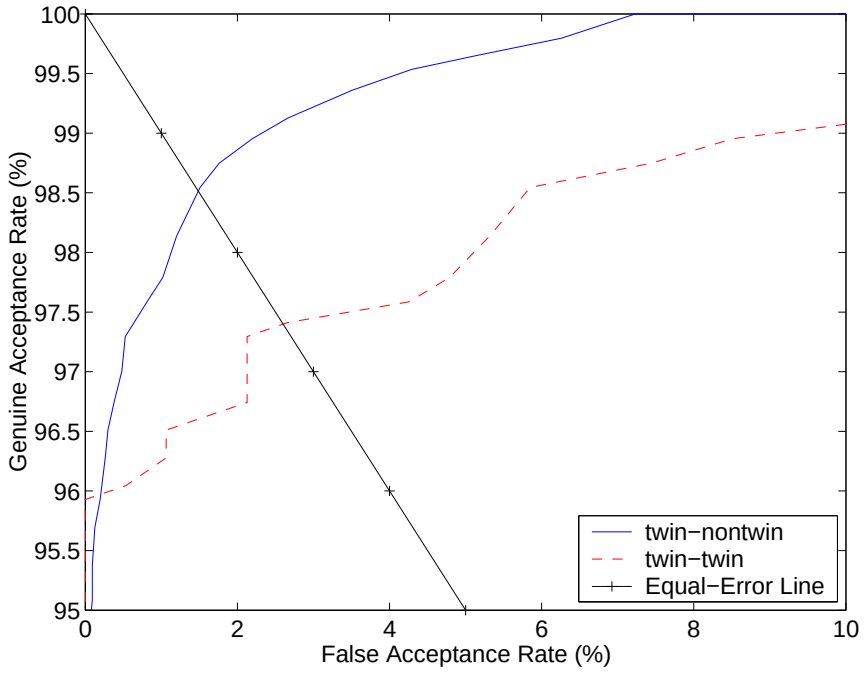
A randomly chosen subset of the rolled identical twin fingerprints collected for the National Heart, Lung, and Blood Institute (NHLBI) twin study [66, 168] is used in our experiments. The fingerprints were acquired using the methods documented in [169]. The fingerprints of the index fingers of 100 pairs of identical twins were scanned using an IBM flatbed color scanner in grayscale mode at 500 *dpi* resolution. Some of the original fingerprints were in ink while others were taken on a sensitized paper with ink-less fluid. The latter tend to fade with time. Due to differences in paper quality and degradation of the print over time, several of these fingerprints are of poor quality. We rejected some of the very poor quality fingerprints and used only 94 pairs of identical twin fingerprints in our study. See Figures 2.3(a) and (b) for examples of fingerprint images in our twin database.

To study the similarity of identical twin fingerprints, we matched every fingerprint in our twin database with every other fingerprint. See Figure 2.3 for an example of minutiae extraction for twin fingerprints. Figures 2.4 and 2.5 show examples of matching twin-nontwin fingerprints, twin-twin fingerprints, and two impressions of the same finger of a person. In Figure 2.6(a), the dash line shows the twin-twin imposter distribution of matching scores computed by matching a fingerprint with his/her identical twin sibling (twin-twin match), while the solid line shows the twin-nontwin imposter distribution of matching scores between a person's fingerprint and everyone else except his/her twin (twin-nontwin match). The twin-twin imposter distribution was estimated using 188 ($94 \times 2$) matchings between the 94 twin fingerprint pairs in our identical twin database whereas the twin-nontwin imposter distribution was estimated using $17,484$ ($94 \times 93 \times 2$) matchings. Figure 2.6(a) shows that the twin-twin imposter distribution is slightly shifted to the right of the twin-nontwin distribution indicating that twin-twin fingerprints are generally more similar than twin-nontwin fingerprints. The twin-twin and twin-nontwin distributions are found to be significantly different (greater than 99.99% confidence) using the Kolmogorov-Smirnov test [179].

The genuine distribution of matching scores is estimated by matching multiple fingerprint images of the same finger. Since we had access to only a single impression of the fingers in our twin database, we had to synthesize the genuine distribution for twin-twin matching. Since the identical twin fingerprint images in our database were obtained by rolling inked fingers of the subjects by fairly experienced finger-printers, we expect the genuine distribution characteristics of the twin database to closely correspond to that obtained from a standard public domain fingerprint database




(a)




(b)

Figure 2.6: (a) Distribution of matching scores for twin-twin imposter, twin-nontwin imposter, and genuine fingerprint matchings. (b) ROC curves for twin-twin and twin-nontwin minutiae pattern matchings.

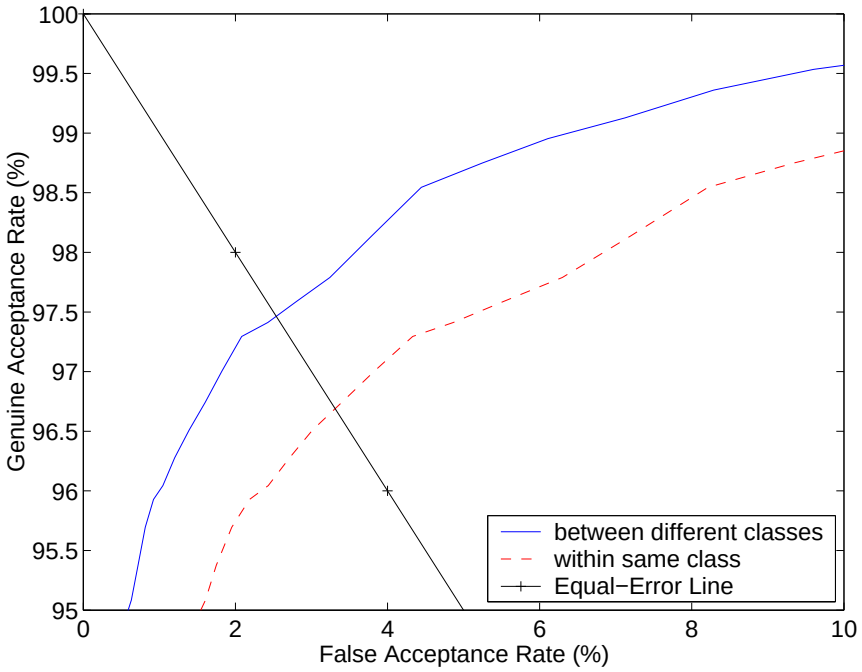



Figure 2.7: Effect of fingerprint class type on the matching score.

(e.g., NIST9 CD No. 1) [42]. This database, consisting of 1,800 fingerprint images taken from 900 independent fingers, two impressions per finger, was used to compute the genuine distribution which is shown in Figure 2.6(a). This genuine distribution along with the two "imposter" distributions in Figure 2.6(a) were used to generate the Receiver Operating Characteristics (ROC) [172, 92] curves shown in Figure 2.6(b). Figure 2.6(b) shows that, due to the similarity of twin fingerprints, the ability of the system to distinguish identical twins is lower than its ability to distinguish twin-nontwin pairs. However, contrary to claims made in the popular press [53], the automatic fingerprint identification system can still be used to distinguish between identical twins without a drastic degradation in performance. See Figure 2.2 for an illustration. Table 2.1 shows the trade-off between FARs and FRRs of twin-twin and twin-nontwin matchings for different thresholds on the matching score.

To quantify the performance degradation of a fingerprint verification system due to the inherent twin-twin similarity in fingerprints, we assume that twin-nontwin imposter distribution is representative of the matchings between unrelated people (nontwins). Suppose a fingerprint verification was set to operate at a decision threshold of $T$ to satisfy the specified FAR requirements. Now, suppose that identical twins use this automatic fingerprint identification system. Since the twin-twin imposter distribution in Figure 2.6(a) is slightly to the right of the twin-nontwin distribution, this will increase the FAR of the system but will have no effect on the FRR. The FAR for identical twins is generally 2% to 6% higher than twin-nontwin matchings depending on the system operating point (different thresholds). The quantitative implication of this in the performance of a fingerprint matching system is as follows. Suppose our system is developed on fingerprints of unrelated people (nontwins) and is set to operate at, say, a threshold of 20 which corresponds to an FAR of $\sim 1\%$ (see row 2 of Table 2.1). Now, if 1 million unrelated people (nontwins) used the system, then, based on our empirical distributions, $10,000$ people will be falsely accepted while $22,000$ people will be falsely rejected. However, if $500,000$ identical twin pairs (1 million twins) used the system operating at the same threshold of 20, then $48,000$ of these will be falsely accepted while $22,000$ people will be falsely rejected. Notice the increase in the false acceptance rate from 1.02% to 4.79%.

To safeguard against twin fraud, we can set the operating point of our system pessimistically at a threshold of 26 which corresponds to an FAR of $\sim 1\%$ for twin-twin matchings and an FAR of $\sim 0.3\%$ for twin-nontwin matchings. This raises the FRR to $\sim 3.5\%$ as opposed to 2.2% when operating at a threshold of 20. This

Table 2.1: False accept and false reject rates with different threshold values for the twin database.

| Threshold T | FRR (%) | FAR (twin-twin) (%) | FAR (twin-nontwin) (%) |
|---|---|---|---|
| 16 | 1.05 | 8.51 | 2.20 |
| 20 | 2.20 | 4.79 | 1.02 |
| 24 | 3.00 | 2.13 | 0.48 |
| 26 | 3.49 | 1.06 | 0.29 |

means that in the worst case scenario (when all the people accessing the system are twins), the system will falsely accept 10,000 people out of one million at the expense of falsely rejecting 35,000 people. In the best case (when there are no twins accessing the system), only $3,000$ people will be falsely accepted while falsely rejecting $35,000$ people. In practice, the system will falsely accept between $3,000$ and $10,000$ people (between 0.3% and 1% ), depending upon the fraction of twins in our sample population of 1 million while falsely rejecting $35,000$ people.

Dermatoglyphics studies have suggested that there is a large correlation between the fingerprint types of identical twins. To confirm this claim, we manually classified the 94 pairs of identical twin fingerprints in our database into five classes (right loop, left loop, whorl, arch, and tented arch). The class correlation between the index fingers of identical twins is found to be 0.775 (fraction of identical twin pairs whose index fingerprints have the same class label). The natural proportion of occurrence of each of the five major classes of fingerprints in the index finger is 0.3252, 0.3648, 0.1703, 0.0616, and 0.0779 for whorl (W), right loop (R), left loop (L), arch (A), and tented arch (T), respectively [173]. If we randomly choose two index fingerprint images from a large database, the probability that these two fingerprints will have

the same class label is equal to $p_W^2 + p_R^2 + p_L^2 + p_A^2 + p_T^2$, i.e., 0.2718, where $p_W$, $p_R$, $p_L$, $p_A$, and $p_T$, are the probabilities of a fingerprint chosen at random belonging to the class of whorl, right loop, left loop, arch, and tented arch, respectively. Thus, there is only 0.2718 chance that two randomly chosen index fingers will have the same type which is much lower than the 0.775 chance that the fingerprints of two identical twins will have the same class label.

We believe that the global similarity of fingerprints (shown as class similarity) is, to a certain extent, responsible for the local similarity (shown in the matching performance). Consider two fingerprints that belong to the same class (e.g., right loop). Since the minutiae can exist only along the ridges (although at random locations), the matching score between these two fingerprints is likely to be higher than the matching score between two sets of random point patterns. To study the correlation of class information with the matching performance, we used the NIST4 database [41] which has $4,000$ fingerprint images collected from $2,000$ independent fingers with 800 fingerprints from each of the five classes.

We computed the genuine distribution from $3,600$ matchings between the two impressions of the same finger from $1,800$ good quality fingerprint pairs from the NIST4 database. The between-class and within-class distributions were computed from about $130,000$ matchings each. The ROCs for between-class and within-class matchings are shown in Figure 2.7. Note that the matching performance is better for fingerprints belonging to different classes compared to fingerprints belonging to the same class. Also, the magnitude of the shift between the two ROCs in Figure 2.7 is of the same order of magnitude as the one manifested in Figure 2.6(b). Thus, we

have shown that the minutiae-based similarity in identical twin fingerprints, is of the same order as the similarity between unrelated people who have the same fingerprint class label. Hence, the larger similarity observed in identical twins is due to the high class correlation in their fingerprint types.

### 2.1.3 Summary

One out of every eighty births results in twins and one third of all the twins are monozygotic (identical) twins [86]. Some identical twins have been reported to be involved in fraud, which can be called "twin fraud", since people mistake the identities of the identical twins. The childhood mischief by the identical twins of switching places on their teachers and taking each other's exams may grow into serious criminal activities in adulthood such as buying a single insurance for identical twin siblings or claiming welfare benefits twice when only one sibling is unemployed. There have been cases reported where an identical twin was sentenced for a crime that was committed by his/her sibling [53]. Fertility treatments have resulted in an increase in the identical twin birth rate (in fact, according to a study by Robert Derom [53], the identical twin birth rate is about twice as high for women who use fertility drugs). Further, because of the medical advances in the treatment of premature babies, the population of identical twins is increasing.

We have shown that even though identical twin fingerprints have large class correlation, they can still be distinguished using a minutiae-based automatic fingerprint identification system; though with a slightly lower accuracy than nontwins. Our re-

sults suggest that the marginal degradation in performance may be related to the dependence of the minutiae distribution on fingerprint class.

What are the implications of our empirical results in person identification applications? In authentication applications, marginal degradation in accuracy performance will have almost no effect on "evil" twins posing as impostors. In large scale fingerprint based identification applications, a small degradation in authentication accuracy may imply a significant degradation in the recognition accuracy. Further, if the degradation in the performance is dependent on the class correlation which in turn depends on the genetic constitution (as suggested by the dermatoglyphics studies), it may imply that benefits reaped by composition of ten-finger information may have been overestimated in the literature. Further, the magnitude of performance degradation of a minutiae-based fingerprint matcher may depend upon the genetic relationship among a target population corpus. Both of these effects may need further investigation; more research is necessary for class-independent minutiae-based matchers. Since the accuracy performance of a minutiae-based fingerprint matcher degrades with genetic similarity in the population, alternate independent representations of fingerprints should be explored that can be combined with the minutiae representation to yield a more accurate automatic fingerprint matching system. Finally, fingerprint classification applications used for the binning of population to increase efficiency of fingerprint based search may not be very efficient in genetically related population.

## 2.2 Environmental Factors

### 2.2.1 Introduction

Our interest in the fingerprint individuality problem is twofold. Firstly, a scientific basis (reliable statistical estimate of the matching error) for fingerprint comparison can determine the admissibility of fingerprint identification in the courts of law as an evidence of identity. Secondly, it can establish an upper bound on the performance of an automatic fingerprint verification system. Here, we develop a fingerprint individuality model that attempts to estimate the probability of a false association. We use this model to establish an upper bound on the performance of a fingerprint verification system [11].

In order to solve the individuality problem, we need to first define *a priori* the representation of a fingerprint (*pattern*) and the metric for the similarity. Fingerprints can be represented by a large number of features, including the overall ridge flow pattern, ridge frequency, location and position of singular points (core(s) and delta(s)), type, direction, and location of minutiae points, ridge counts between pairs of minutiae, and location of pores (see Figures 2.8(a) and (b)). All these features contribute in establishing fingerprint individuality. In this study, we have chosen minutiae representation of the fingerprints because it is utilized by forensic experts, has been demonstrated to be relatively stable and has been adopted by most of the automatic fingerprint matching systems.

Given a representation scheme and a similarity metric, there are two approaches for determining the individuality of the fingerprints. In the empirical approach, *repre-*

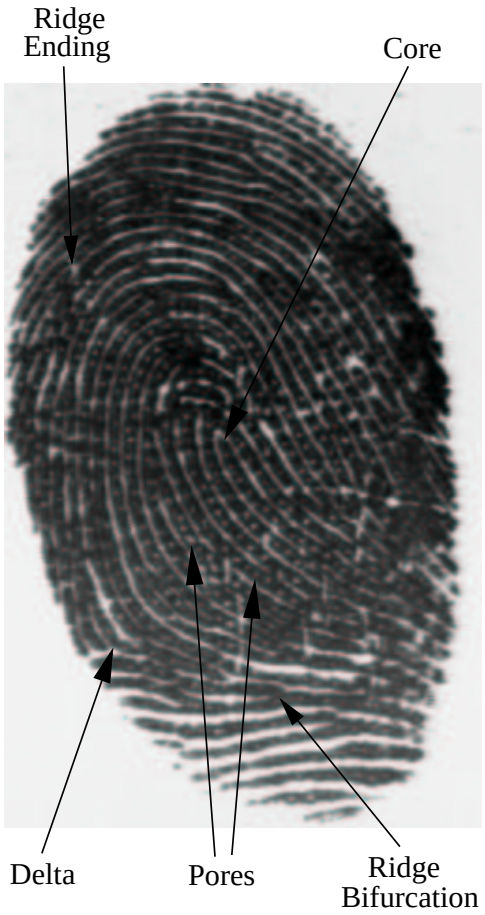



Figure 2.8: A fingerprint image of type "right loop". The overall ridge structure, singular points, and sweat pores are shown.

*sentative* samples of fingerprints are collected and using a *typical* fingerprint matcher, the accuracy of the matcher on the samples provides an indication of the uniqueness of the fingerprint with respect to the matcher. There are known problems (and costs) associated with collection of the *representative* samples. In a theoretical approach to individuality estimation, one models all realistic phenomenon affecting inter-class and intra-class pattern variations. Given the similarity metric, one could then, theoretically estimate the probability of a false association. Theoretical approaches are often limited by the extent to which the assumed model conforms to the reality. In this work, we emphasize the theoretical formulation of the fingerprint individuality model

based on a number of parameters derived from a database of fingerprint images. We also juxtapose the probabilities obtained from individuality model with the empirical matcher accuracy results.

Minutiae patterns are generated by the underlying fingerprints which are smoothly flowing oriented textures. The minutiae points are not randomly distributed since the positions are determined by the ridges (see Figure 2.9). Further, the orientations of nearby minutiae are strongly correlated. Thus, the configuration space spanned by the minutiae pattern is smaller than that spanned by a pattern of (directed) random points. This typically implies that the probability of finding sufficiently similar prints from two different fingers is higher than that of finding sufficiently similar sets of random (directed) point patterns. In our study, we have imposed realistic fingerprint structural (e.g., ridge/valley position, ridge orientation) constraints on a random point configuration space to derive a more effective estimate of the probability of false association.

The total number of degrees-of-freedom of the pattern space (e.g., minutiae configuration space) does not directly relate to the discriminability of the different patterns (e.g., minutiae from different fingers). The effective estimation of discriminatory information can only be achieved by taking into account intra-pattern variations [16]. There are several sources of variability in the multiple impressions of a finger [11]: non-uniform contact (with the sensor), irreproducible contact, inconsistent contact, and imaging artifacts. This variability in multiple impressions of a finger manifests itself in (*i*) detection of spurious minutiae or missing genuine minutiae, (*ii*) displacement/disorientation (also called deformation) of the genuine detected minutiae, and

(*iii*) transformation of the type of minutiae (connective ambiguity). This entails designing a similarity metric (matcher) that accommodates these intra-class variations. As a result, the probability of false association increases significantly.

Most of the earlier approaches did not explicitly incorporate these (intra-class) variabilities into their individuality models (see [47] for a critical review of several models) and, therefore, overestimate the fingerprint individuality. Since most of the existing models of individuality do not address the problems associated with occurrence of spurious minutiae or missing genuine minutiae, they do not provide a systematic framework to address issues related to a partial representational match between two fingerprints (e.g., what is the probability of finding 7 matched minutiae in two fingerprints with 18 and 37 minutiae, respectively?). This is very important in an automatic fingerprint matching system (feature extraction algorithms are not as accurate as a well-trained fingerprint expert in detecting minutiae) and in matching latents (where a print depicting a small portion of a finger is matched against a print depicting a full finger). Although, in a manual fingerprint matching procedure, the likelihood of *detecting* false minutia is significantly smaller than that in an automatic system, the prints imaged from different portions of fingers may give rise to the variability in the number of detected minutia. Our approach not only explicitly models the situation of partial representational match but also incorporates constraints on the configuration space due to intra-pattern variations (e.g., number of minutia, minutia position/orientation, image area) based on empirical estimates derived from the ground truth data marked on fingerprints obtained in a realistic environment.

The rest of the Chapter is organized as follows. Section 2.2.2 presents a summary

of major fingerprint individuality studies and compares the probability of a fingerprint configuration obtained by different models. Section 2.2.3 presents the proposed fingerprint individuality model, and section 2.2.4 presents the results. Summary and discussions are presented in section 2.2.5.

### 2.2.2 Background

The early individuality studies typically focused on a predominantly minutiae-based representation; some studies explicitly factored in fingerprint class (e.g., right loop, left loop, whorl, arch, tented arch, etc.) information. The type, direction, and location of minutiae were the most commonly used features in the early individuality studies. See Table 2.2 for a comparison of the features used in fingerprint individuality models. The types of minutiae used varies from one study to other: some studies used two minutia types (ending and bifurcation) whereas others used as many as 13 types of events (e.g., empty cell, ridge ending, ridge fork, island, dot, broken ridge, bridge, spur, enclosure, delta, double fork, trifurcation, multiple events) [94]. Later models considered additional features to determine the probability of occurrence of a particular fingerprint configuration (e.g., ridge counts [47], sweat pores [29]).

Most of the early individuality studies examined the distinctiveness of a portion/feature of the fingerprint. Under simplifying assumptions (e.g., implicit assumptions about statistical independence of events and the corresponding event distributions are identical), these studies estimated the distinctiveness of the entire fingerprint (total pattern variation) by collating the distinctiveness in the feature extracted from

fingerprints (total feature variation). We will refer to these total pattern variation-based fingerprint individuality estimates as the *probability of fingerprint configuration*. A summary of these studies is presented below.

The fingerprint individuality problem was first addressed by Galton in 1892 [70], who considered a square region spanning six-ridges in a given fingerprint. He assumed that, on an average, a fingerprint can be covered by 24 such six-ridge wide independent square regions. Galton estimated that he could correctly reconstruct any of the regions with a probability of $\frac{1}{2}$, by looking at the surrounding ridges. Accordingly, the probability of a specific fingerprint configuration, given the surrounding ridges is $\left(\frac{1}{2}\right)^{24}$. He multiplied this conditional (on surrounding ridges) probability with the probability of finding the surrounding ridges to obtain the probability of occurrence of a fingerprint as

$$P(\text{Fingerprint Configuration}) = \frac{1}{16} \times \frac{1}{256} \times \left(\frac{1}{2}\right)^{24} = 1.45 \times 10^{-11}, \qquad (2.1)$$

where $\frac{1}{16}$ is the probability of occurrence of a specific fingerprint type (such as arch, tented arch, left loop, right loop, double loop, whorl, etc.) and $\frac{1}{256}$ is the probability of occurrence of the correct number of ridges entering and exiting each of the 24 regions. Eq. (2.1) gives the probability that a particular fingerprint configuration in an average size fingerprint (containing 24 regions defined by Galton) will be observed in nature. Roxburgh [170], Pearson [98], and Kingston [39] objected to Galton's assumption that the probability of occurrence of any particular ridge configuration in a six-ridge square is $\frac{1}{2}$, and claimed that Eq. (2.1) grossly underestimated the

fingerprint individuality (i.e., overestimated the probability of occurrence). Pearson [98] argued that there could be 36 ($6 \times 6$) possible minutiae locations within one of Galton's six-ridge-square regions, leading to a probability of occurrence of a particular fingerprint configuration of

$$P(\text{Fingerprint Configuration}) = \frac{1}{16} \times \frac{1}{256} \times \left(\frac{1}{36}\right)^{24} = 1.09 \times 10^{-41}. \tag{2.2}$$

A number of subsequent models (Henry [64], Balthazard [176] (cf. [47]), Bose [47], Wentworth and Wilder [36], Cummins and Midlo [79], and Gupta [161]) are interrelated and are based on a fixed probability, $p$, for the occurrence of a minutiae. They compute the probability of a particular $N$-minutiae fingerprint configuration as

$$P(\text{Fingerprint Configuration}) = p^N. \tag{2.3}$$

In the following, we provide the values of $p$ used in various studies. In most cases, the authors do not present any details on how they arrived at their choice of $p$.

- Henry [64] chose $p = \frac{1}{4}$ and added 2 to the number of minutiae, $N$, if the fingerprint type and core-to-delta ridge count could be determined from the given (latent) fingerprint.

- Balthazard [176] also set $p = \frac{1}{4}$, under the assumption that there are four types of equally likely minutiae events: (*i*) fork (bifurcation) to the right, (*ii*) fork to the left, (*iii*) ending to the right, and (*iv*) ending to the left.

- Bose [47] adopted $p = \frac{1}{4}$, under the assumption that there are four possibilities

in each square region of one ridge-interval width in a fingerprint: (*i*) a dot, (*ii*) a fork, (*iii*) an ending, and (*iv*) a continuous ridge.

- Wentworth and Wilder [36] chose $\frac{1}{50}$ as the value of $p$.

- Cummins and Midlo [79] adopted the same value of $p$ as Wentworth and Wilder, but introduced a multiplicative constant of $\frac{1}{31}$ to account for the variation in fingerprint pattern type.

- Gupta [161] estimated the value of $p$ as $\frac{1}{10}$ for forks and endings, and $\frac{1}{100}$ for the less commonly occurring minutiae types, based on 1,000 fingerprints. He also used a fingerprint-type-factor of $\frac{1}{10}$ and correspondence-in-ridge-count-factor of $\frac{1}{10}$.

Because of the widely varying values of $p$ used in the above studies, the probability of a given fingerprint configuration also dramatically varies from one model to the other.

Roxburgh [170] proposed a more comprehensive analysis to compute the probability of a fingerprint configuration. His analysis was based on considering a fingerprint as a pattern with concentric circles, one ridge interval apart, in a polar coordinate system. Roxburgh also incorporated a quality measure of the fingerprint into his calculations. He computed the probability of a particular fingerprint configuration to be:

$$P(\text{Fingerprint Configuration}) = \left(\frac{C}{P}\right)\left(\frac{Q}{RT}\right)^{N}, \tag{2.4}$$

where $P$ is the probability of encountering a particular fingerprint type and core type,

$Q$ is a measure of quality ($Q = 1.5$ for an average quality print, $Q = 3.0$ for a poor quality print), $R$ is the number of semicircular ridges in a fingerprint ($R = 10$), $T$ is the corrected number of minutiae types ($T = 2.412$), and $C$ is the number of possible positions for the configuration ($C = 1$). Amy [102] (cf. [47]) considered the variability in minutiae type, number, and position in his model for computing the probability of a fingerprint configuration. He further recognized that $K$ multiple comparisons of the fingerprint pair (e.g., each hypothesized orientation alignment, each reference point correspondence) increase the possibility of false association which is given by

$$P(\text{False Association}) = 1 - (1 - P(\text{Fingerprint Configuration}))^K . \tag{2.5}$$

Kingston's [39] model, which is very similar to Amy's model, computes the probability of a fingerprint configuration based on the probabilities of the observed number of minutiae, observed positions of minutiae, and observed minutiae types as follows:

$$P(\text{Fingerprint Configuration}) = (e^{-y})(y^N/N!)(P_1)\prod_{i=2}^{N}(P_i)\frac{(0.082)}{[S-(i-1)(0.082)]}, \tag{2.6}$$

where $y$ is the expected number of minutiae in a region of given size $S$ (in mm$^2$) and $P_i$ is the probability of occurrence of a particular minutiae type.

Most of the models discussed above implicitly assume that fingerprints are being matched manually. The probability of observing a given fingerprint feature is estimated by manually extracting the features from a small number of fingerprint images. Champod and Margot [37] used an AFIS to extract minutiae from 977 fingerprint im-

Table 2.2: Fingerprint features used in different models.

| Author | Fingerprint features used |
|---|---|
| Galton (1892) | ridges, minutiae types |
| Pearson (1930) | ridges, minutiae types |
| Henry (1900) | minutiae locations,<br>types, core-to-delta ridge count |
| Balthazard (1911) | minutiae locations, two types,<br>and two directions |
| Bose (1917) | minutiae locations and three types |
| Wentworth & Wilder (1918) | minutiae locations |
| Cummins & Midlo (1943) | minutiae locations, types,<br>core-to-delta ridge count |
| Gupta (1968) | minutiae locations and types,<br>types, ridge count |
| Roxburgh (1933) | minutiae locations, two minutiae types,<br>two orientations, fingerprint and core types,<br>number of possible positionings, area,<br>fingerprint quality |
| Amy (1948) | minutiae locations, number,<br>types, and orientation |
| Trauring (1963) | minutiae locations,<br>two types, and two orientations |
| Kingston (1964) | minutiae locations,<br>number, and types |
| Osterburg et al. (1980) | minutiae locations and types |
| Stoney et al. (1986) | minutiae locations,<br>distribution, orientation, and types, variation<br>among prints from the same source, ridge<br>counts, and number of alignments |

Table 2.3: Comparison of probability of a particular fingerprint configuration using different models. For a fair comparison, we do not distinguish between minutiae types. By assuming that an average size fingerprint has 24 regions ($R = 24$) as defined by Galton, 72 regions ($M = 72$) as defined by Osterburg et al., and has 36 minutiae on an average ($N = 36$), we compare the probability of observing a given fingerprint configuration in the third column of the table. The probability of observing a fingerprint configuration with $N = 12$, and equivalently, $R = 8$, is given in braces in the third column. Note that all probabilities represent a full ($N$ minutiae) match as opposed to a partial match (see Table 2.5).

| Author | P(Fingerprint Configuration) | N=36,R=24,M=72 (N=12,R=8,M=72) |
|---|---|---|
| Galton (1892) | $\frac{1}{16} \times \frac{1}{256} \times \left(\frac{1}{2}\right)^R$ | $1.45 \times 10^{-11}$ ($9.54 \times 10^{-7}$) |
| Pearson (1930) | $\frac{1}{16} \times \frac{1}{256} \times \left(\frac{1}{36}\right)^R$ | $1.09 \times 10^{-41}$ ($8.65 \times 10^{-17}$) |
| Henry (1900) | $\left(\frac{1}{4}\right)^{N+2}$ | $1.32 \times 10^{-23}$ ($3.72 \times 10^{-9}$) |
| Balthazard (1911) | $\left(\frac{1}{4}\right)^N$ | $2.12 \times 10^{-22}$ ($5.96 \times 10^{-8}$) |
| Bose (1917) | $\left(\frac{1}{4}\right)^N$ | $2.12 \times 10^{-22}$ ($5.96 \times 10^{-8}$) |
| Wentworth & Wilder (1918) | $\left(\frac{1}{50}\right)^N$ | $6.87 \times 10^{-62}$ ($4.10 \times 10^{-21}$) |
| Cummins & Midlo (1943) | $\frac{1}{31} \times \left(\frac{1}{50}\right)^N$ | $2.22 \times 10^{-63}$ ($1.32 \times 10^{-22}$) |
| Gupta (1968) | $\frac{1}{10} \times \frac{1}{10} \times \left(\frac{1}{10}\right)^N$ | $1.00 \times 10^{-38}$ ($1.00 \times 10^{-14}$) |
| Roxburgh (1933) | $\frac{1}{1000} \times \left(\frac{1.5}{10 \times 2.412}\right)^N$ | $3.75 \times 10^{-47}$ ($3.35 \times 10^{-18}$) |
| Trauring (1963) | $(0.1944)^N$ | $2.47 \times 10^{-26}$ ($2.91 \times 10^{-9}$) |
| Osterburg et al. (1980) | $(0.766)^{M-N}(0.234)^N$ | $1.33 \times 10^{-27}$ ($3.05 \times 10^{-15}$) |
| Stoney (1985) | $\frac{N}{5} \times 0.6 \times (0.5 \times 10^{-3})^{N-1}$ | $1.2 \times 10^{-80}$ ($3.5 \times 10^{-26}$) |

ages scanned at a relatively high resolution of 800 *dpi*. They generated frequencies of minutiae occurrence and minutiae densities after manually verifying the thinned ridges produced by the AFIS to ensure that the feature extraction algorithm did not introduce errors. They considered minutiae only in concentric bands (five ridges wide) above the core and acknowledged that their individuality estimates were conservative (i.e., provided an upper bound). As an example, they estimated the probability of occurrence of a seven-minutiae configuration (five endings and two bifurcations) as $2.25 \times 10^{-5}$.

Osterburg et al. [94] divided fingerprints into discrete cells of size $1\ mm \times 1\ mm$. They computed the frequencies of 13 types of minutiae events (including an empty cell) from 39 fingerprints (8,591 cells) and estimated the probability that 12 ridge endings will match between two fingerprints based on an average fingerprint area of 72 $mm^2$ as $1.25 \times 10^{-20}$. Sclove [157] modified Osterburg et al.'s model by incorporating the observed dependence of minutiae occurrence in cells and came up with an estimate of probability of fingerprint configuration that is slightly higher than that obtained by Osterburg et al. Stoney and Thornton [47] criticized Osterburg et al.'s and Sclove's models because these models did not consider the fingerprint ridge structure, distortions, and the uncertainty in the positioning of the grid. Stoney and Thornton [47] critically reviewed earlier fingerprint individuality models and proposed a detailed set of fingerprint features that should be taken into consideration. These features included ridge structure and description of minutiae location, ridge counts between pairs of minutiae, description of minutiae distribution, orientation of minutiae, variation in minutiae type, variation among fingerprints from the same source,

number of positions (different translations and rotations of the input fingerprint to match with the template), and number of comparisons performed with other fingerprints for identification.

Stoney's [49] model is different from other models in that it attempts to characterize a significant component of pairwise minutiae dependence. Stoney [49] and Stoney and Thornton [47] studied probabilities of occurrences of various types of minutiae, their orientation, number of neighboring minutiae, and distances/ridge counts to the neighboring minutiae. Given a minutiae set, they calculated the probability of a minutiae configuration by conjoining the probabilities of the individual events in the configuration. For instance, they proposed a linear ordering of minutiae in a minutia configuration and recursively estimated the probability of a $n$-minutiae configuration from the probability of a $(n-1)$-minutiae configuration and the occurrence of a new minutiae of certain type/orientation at a particular distance/ridge counts from its nearest minutiae within the $(n-1)$-minutiae configuration. The model also incorporated constraints due to connective ambiguity and due to minutia-free areas. The model corrected for the probability of false association by accounting for the various possible linear orderings which could initiate/drive the search for correspondence. A sample calculation for computing the probability of a false association using Stoney's model is given below.

$$\begin{aligned} P(\text{False Association}) &= 1 - \left(1 - 0.6 * \left(0.5 \times 10^{-3}\right)^{(N-1)}\right)^{\lfloor \frac{N}{5} \rfloor} \\ &\approx \frac{N}{5} \times 0.6 * \left(0.5 \times 10^{-3}\right)^{(N-1)}. \end{aligned} \tag{2.7}$$

For the sake of simplicity, we have considered only a rudimentary version of Stoney's model for the above computation; it is arbitrarily assumed that the probability of a typical *starting* minutia is 0.6, a typical neighboring minutia places an additional constraint of $5 \times 10^{-3}$ on the probability, and there are no constraints due to connective ambiguity, minutia-free areas or minutia-free borders are assumed. Finally, it is (arbitrarily) assumed that one in every five minutia can potentially serve as a starting point for a new search. We believe that a more realistic estimation of the individuality based on Stoney's model would not deviate from the simplistic estimation presented here by more than a couple of orders of magnitude.

Stoney and Thornton identified weaknesses in their model and acknowledged that one of the most critical requirements, i.e., consideration of variation among prints from the same source, is not sufficiently addressed in their model. Their tolerances for minutiae position were derived from successive printings under ideal conditions and are far too low to be applicable in actual fingerprint comparisons.

The models discussed above (including Amy's model of false association due to multiple comparisons) concentrated mainly on measuring the amount of detail in a single fingerprint (i.e., estimation of the probability of a fingerprint configuration). These models did not emphasize the intra-class variations in multiple impressions of a finger. We will refer to the quantifications of fingerprint individuality which explicitly consider the intra-class variations as the *probability of correspondence*. Trauring [125] was the first to concentrate explicitly on measuring the amount of detail needed to establish correspondence between two prints from the same finger using an AFIS and observed that corresponding fingerprint features could be displaced from each other

by as much as 1.5 times the inter-ridge distance. He further assumed that $(i)$ minutiae are distributed randomly, $(ii)$ there are only two types of minutiae (ending and bifurcation), $(iii)$ the two types of minutiae are equally likely, $(iv)$ the two possible orientations of minutiae are equally likely, and $(v)$ minutiae type, orientation, and position are independent variables. Trauring computed the probability of a coincidental correspondence of $N$ minutiae between two fingerprints to be:

$$P(\text{Fingerprint Correspondence}) = (0.1944)^N. \tag{2.8}$$

Stoney and Thornton's [47] criticism of the Trauring model is that he did not consider ridge count, connective ambiguity, and correlation among minutiae location. Further, they claim that Trauring's assumption that the minutiae types and orientations are equally probable is not correct. The probabilities of observing a particular minutiae configuration from different models are compared in Table 2.3.

There have been a few studies which empirically estimate the probability of finding a fingerprint in a large database that successfully matches the input fingerprint. Meagher et al. [151] (for more details see Stiles [123]) matched about 50,000 rolled fingerprints belonging to the same fingerprint class (left loop) with each other to compute the impostor distribution. However, the genuine distribution was computed by matching each fingerprint image with itself; this ignores the variability present in different impressions of the same finger. Further, they assumed that the impostor and the genuine distributions follow a Gaussian distribution and computed the probability of a false association to be $10^{-97}$. This model grossly underestimates the probability

of a false association because it does not consider realistic intra-class variations in impressions of a finger (see also, Stoney et al. [47] and Wayman [91]).

### 2.2.3 A Model of Fingerprint Individuality

We have developed a model to obtain a realistic and more accurate probability of correspondence between fingerprints. The probabilities obtained using this model will be compared against empirical values using an *automatic fingerprint matching system* (AFMS) [11] (an AFIS is used for identification; an AFMS is used for verification). To estimate the probability of correspondence, we make the following assumptions:

1. We consider only minutiae features since ($i$) most of the discriminatory power of the AFMS is based on minutiae features, and ($ii$) for an objective measurement of individuality, it is necessary that the representation be consistently reproducible, easily localized, and quantified. Minutiae features have been shown to be stable and practical systems have demonstrated a reliable extraction of minutia representation from fingerprints of reasonable image quality. Only ridge endings and ridge bifurcations are considered because the occurrence of other minutiae types such as islands, dots, enclosures, bridges, double bifurcations, trifurcations, etc. is relatively rare. Additionally, we do not distinguish between the two types of minutiae because ridge endings and ridge bifurcations can not be accurately discriminated. Since minutiae can reside only on ridges which follow certain overall patterns in a fingerprint, the minutiae directions are not completely independent of the minutiae locations. We implicitly model the

statistical dependence between minutiae directions and locations in our model. Finally, we have not considered the pairwise minutiae features such as ridge counts in the present analysis.

2. We assume a uniform distribution of minutiae in a fingerprint with the restriction that two minutiae cannot be very close to each other. While minutiae locations are not uniformly distributed, our assumption approximates the slightly overdispersed uniform distribution found by Stoney [48]. Sclove [157] showed that the minutiae tend to cluster. We have not explicitly modeled the clustering tendency of minutiae. Therefore, the assumption of independence of minutiae locations will bias the estimate of the probability of a false association towards higher values. However, it is a common practice in fingerprint individuality studies to make conservative (higher) estimates of the probability of correspondence. Both Sclove [157] and Osterburg et al. [94] discuss how these conservative estimates favor a suspect in a criminal investigation, in the sense that it gives the suspect the benefit of the doubt by lowering the certainty attached with the fingerprint matching.

3. Correspondence of a minutiae pair is an independent event and each correspondence is equally important. Fingerprint matching systems weigh different correspondence based on their position (e.g., correspondences involving minutiae from peripheral pattern area are weighted less than those minutiae located in the center of the fingerprint). Similarly, it is possible to weight spatially diverse correspondences more than all correspondences localized in a narrow

spatial neighborhood. Our analysis currently ignores such dependencies among the minutiae correspondences.

4. We do not explicitly take into account fingerprint image quality in individuality determination. It is very difficult to reliably assign a quality index to a fingerprint because image quality is a subjective concept. Our approach to incorporating image quality in fingerprint matching assumes that only a subset of the true minutiae in a fingerprint will be detected. All correspondences are considered reliable and no certainty is associated with a correspondence based on the fingerprint image quality. In good quality fingerprints, one could use conflicting evidence (when a minutia in input does not match any minutiae in template) to reject the hypothesis that the input and the template fingerprints are the same. However, there will be some errors in identifying minutiae in fingerprints with poor quality. Therefore, we explicitly consider only the positive evidence from a minutiae correspondence; the negative information from the conflicting evidence (e.g., a minutia that does not match) is ignored.

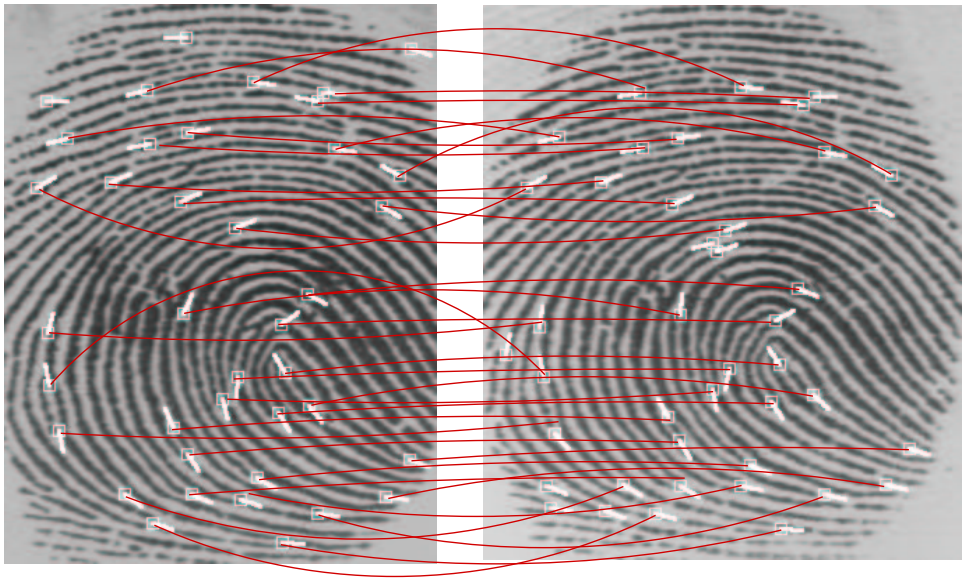
5. Ridge widths are same across the population and spatially uniform in the same finger. This assumption is justified because the pressure variations could make non-uniform ridge variations uniform and vice versa. Further, there may be only limited discriminatory information in the ridge frequency.

6. The analysis of matchings of different impressions of the same finger binds the parameters of the probability of matching minutiae in two fingerprints from different fingers.

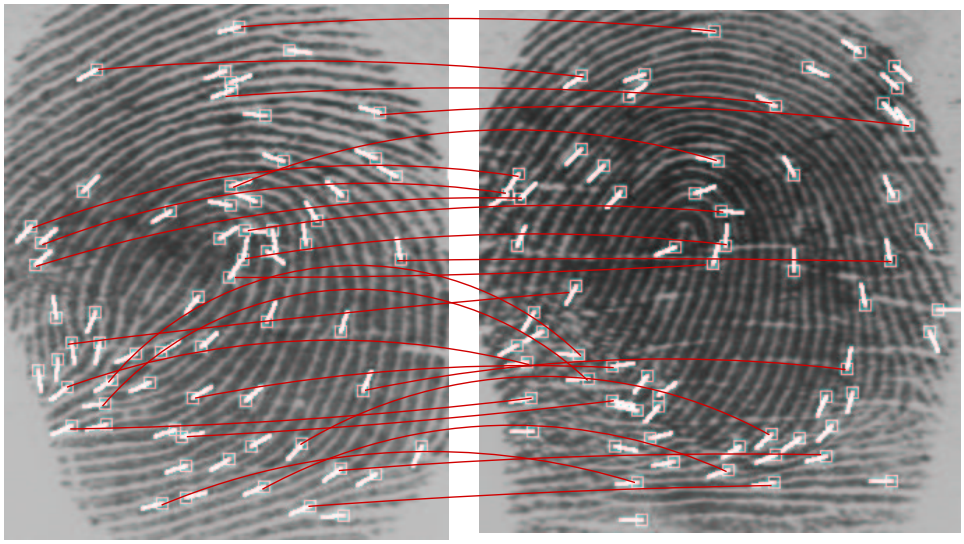
7. We assume that there exists one and only one alignment between the template and the input minutiae sets.

The fingerprint correspondence problem involves matching two fingerprints; one is called the *template* (stored in the system) and the other is called the *input* (which needs to be verified). We assume that a reasonable *alignment* has been established between the template and the input. The alignment of the input minutiae set with the template minutiae set is done so that the minutiae correspondences can be determined in a small tolerance. In manual fingerprint matching, this alignment is typically based on aligning the fingerprint singularities (core(s) and delta(s)) and ridges. An automatic system may seek an alignment that maximizes a given objective function (such as the number of matching minutiae). This assumption may not be valid when matching a partial (latent) fingerprint with a full print in the database, as there may be several "reasonable" alignments possible. When multiple alignments are indeed warranted by a situation, the probability of false association increases (see Eq. (2.5)).

Given an input fingerprint containing $n$ minutiae, our goal is to compute the probability that any arbitrary fingerprint (template in a database of fingerprints) containing $m$ minutiae will have exactly $q$ corresponding minutiae with the input. Since we only consider fingerprint minutiae which is defined by its location, $(x, y)$, and by the angle of the ridge on which it resides, $\theta$, the input and the template minutiae sets, T and I, respectively, are defined as:



(a)



(b)

Figure 2.9: Automatic minutiae matching. Two impressions of the same finger were matched in (a) 39 minutiae were detected in input (left), 42 in template (right), and 36 "true" correspondences were found. Two different fingers are matched in (b) 64 minutiae were detected in input (left), 65 in template (right), and 25 "false" correspondences were found.

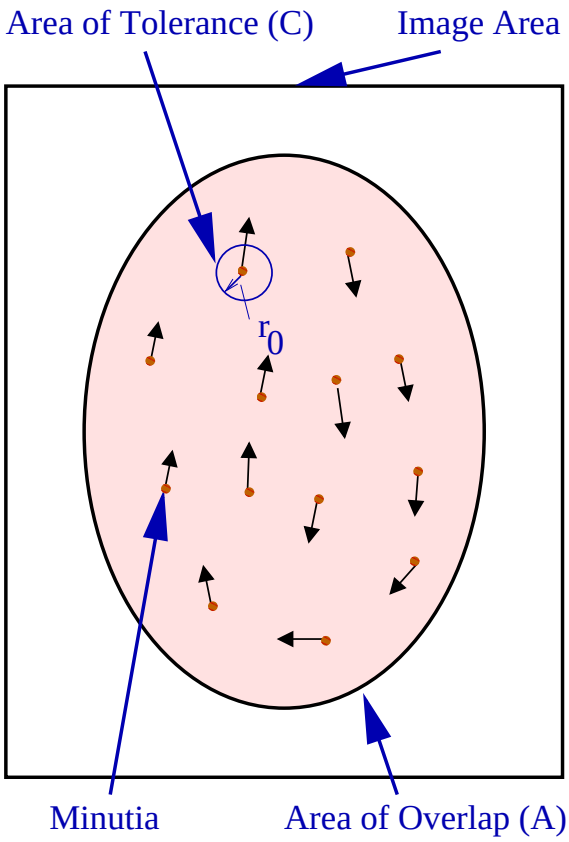



Figure 2.10: Fingerprint and minutiae.

$$T = \{\{x_1, y_1, \theta_1\}, \{x_2, y_2, \theta_2\}, ..., \{x_m, y_m, \theta_m\}\}, \tag{2.9}$$

$$I = \{\{x'_1, y'_1, \theta'_1\}, \{x'_2, y'_2, \theta'_2\}, ..., \{x'_n, y'_n, \theta'_n\}\}. \tag{2.10}$$

Once an alignment between the input minutiae set and the template minutiae set is established, we develop our individuality model. Let us first model the intra-class variation. A minutiae $j$ in the input fingerprint is considered as "corresponding" or

"matching" to the minutiae $i$ in the template, if and only if

$$\sqrt{(x_i' - x_j)^2 + (y_i' - y_j)^2} \leq r_0, \quad \text{and} \tag{2.11}$$

$$\min\left(|\theta_i' - \theta_j|, 360 - |\theta_i' - \theta_j|\right) \leq \theta_0, \tag{2.12}$$

where $r_0$ is the tolerance in distance and $\theta_0$ is the tolerance in angle. Both manual and automatic fingerprint matchings are based on some tolerance both in minutiae location and angle to account for the variations in different impressions of the same finger. Eq. (2.12) computes the minimum of $|\theta_i' - \theta_j|$ and $360 - |\theta_i' - \theta_j|$ because the angles are *mod* 360 (the difference between angles of 2° and 358° is only 4°).

Let $A$ be the total area of overlap between the input and the template fingerprints after a reasonable alignment has been achieved. The probabilities that any arbitrary minutiae in the input will match any arbitrary minutiae in the template, only in terms of location, and only in terms of direction, are given by Eqs. (2.13) and (2.14), respectively. Eq. (2.13) assumes that $(x, y)$ and $(x', y')$ are independent and Eq. (2.14) assumes that $\theta$ and $\theta'$ are independent.

$$P\left(\sqrt{(x_i' - x_j)^2 + (y_i' - y_j)^2} \leq r_0\right) = \frac{\text{area of tolerance}}{\text{total area of overlap}} = \frac{\pi r_0^2}{A} = \frac{C}{A}, \tag{2.13}$$

$$P\left(\min\left(|\theta_i' - \theta_j|, 360 - |\theta_i' - \theta_j|\right) \leq \theta_0\right) = \frac{\text{angle of tolerance}}{\text{total angle}} = \frac{2\theta_0}{360}. \tag{2.14}$$

First we will develop our fingerprint correspondence model when only minutiae locations alone are matched and then introduce the minutiae angles later in the formulation. If the template contains $m$ minutiae, the probability that only one minutia in

the input will correspond to any of the $m$ template minutiae is given by $\frac{mC}{A}$. Now, given two input minutiae, the probability that only the first one corresponds to one of the $m$ template minutiae is the product of the probabilities that the first input minutiae has a correspondence ($\frac{mC}{A}$) and the second minutiae does not have a correspondence ($\frac{A-mC}{A-C}$). Thus, the probability that exactly 1 of the 2 input minutiae matches any of the $m$ template minutiae is $2 \times \frac{mC}{A} \times \frac{A-mC}{A-C}$, since either the first input minutiae alone may have a correspondence or the second input minutiae alone may have a correspondence. If the input fingerprint has $n$ minutiae, the probability that exactly one input minutia matches one of the $m$ template minutiae is

$$p(A, C, m, n) = \begin{pmatrix} n \\ 1 \end{pmatrix} \left(\frac{mC}{A}\right) \left(\frac{A - mC}{A - C}\right). \tag{2.15}$$

The probability that there are exactly $\rho$ corresponding minutiae between the $n$ input minutiae and $m$ template minutiae is then given by:

$$p(A, C, m, n, \rho) = \begin{pmatrix} n \\ \rho \end{pmatrix} \underbrace{\left(\frac{mC}{A}\right) \left(\frac{(m-1)C}{A - C}\right) ... \left(\frac{(m-\rho-1)C}{A-(\rho-1)C}\right)}_{\rho \text{ terms}} \times$$
$$\underbrace{\left(\frac{A - mC}{A - \rho C}\right) \left(\frac{A-(m-1)C}{A-(\rho+1)C}\right) ... \left(\frac{(A-(m-(n-\rho+1))C}{A-(n-1)C}\right)}_{n-\rho \text{ terms}}. \tag{2.16}$$

The first $\rho$ terms in Eq. (2.16) denote the probability of matching $\rho$ minutiae between the template and the input; and remaining $n - \rho$ terms express the probability that $n - \rho$ minutiae in the input do not match any minutiae in the template. Dividing the

numerator and denominator of each term in Eq. (2.16) by $C$, we obtain:

$$p(A,C,m,n,\rho) = \begin{pmatrix} n \\ \\ \rho \end{pmatrix} \left(\frac{m}{\frac{A}{C}}\right) \left(\frac{(m-1)}{\frac{A}{C}-1}\right) ... \left(\frac{(m-\rho-1)}{\frac{A}{C}-(\rho-1)}\right) \times$$
$$\left(\frac{\frac{A}{C}-m}{\frac{A}{C}-\rho}\right) \left(\frac{\frac{A}{C}-(m-1)}{\frac{A}{C}-(\rho+1)}\right) ... \left(\frac{(\frac{A}{C}-(m-(n-\rho+1))}{\frac{A}{C}-(n-1)}\right). \quad (2.17)$$

Letting $M = \frac{A}{C}$, we get

$$p(M,m,n,\rho) = \begin{pmatrix} n \\ \\ \rho \end{pmatrix} \left(\frac{m}{M}\right) \left(\frac{(m-1)}{M-1}\right) ... \left(\frac{(m-\rho-1)}{M-(\rho-1)}\right) \times$$
$$\left(\frac{M-m}{M-\rho}\right) \left(\frac{M-(m-1)}{M-(\rho+1)}\right) ... \left(\frac{(M-(m-(n-\rho-1))}{M-(n-1)}\right). \quad (2.18)$$

By assuming that $M$ is an integer (which is a realistic assumption because $A \gg C$), we can write the above equation in a compact form as:

$$p(M,m,n,\rho) = \frac{n!}{\rho!(n-\rho)!} \times \frac{(M-n)!}{M!} \times \frac{m!}{(m-\rho)!} \times \frac{(M-m)!}{((M-m)-(n-\rho))!}. \quad (2.19)$$

Rearranging the terms,

$$p(M,m,n,\rho) = \frac{m!}{\rho!(m-\rho)!} \times \frac{(M-m)!}{(n-\rho)!((M-m)-(n-\rho))!} \times \frac{(M-n)!n!}{M!}, \quad (2.20)$$

which finally reduces to:

$$p(M, m, n, \rho) = \frac{\begin{pmatrix} m \\ \rho \end{pmatrix} \begin{pmatrix} M-m \\ n-\rho \end{pmatrix}}{\begin{pmatrix} M \\ n \end{pmatrix}}. \tag{2.21}$$

Eq. (2.21) defines a hyper-geometric distribution. To get an intuitive understanding of the probability model for the minutiae correspondence in two fingerprints, imagine that the overlapping area of the template and the input fingerprints is divided into $M$ non-overlapping cells. The shape of the individual cells does not matter, just the number of cells. Now consider a deck of cards containing $M$ distinct cards. Each card represents a cell in the overlapping area. There is one such deck for the template fingerprint and an identical deck for the input fingerprint. If $m$ cards are drawn from the first (template) deck without replacement, and $n$ cards are drawn from the second (input) deck without replacement, the probability of matching exactly $q$ cards among the cards drawn is given by the hyper-geometric distribution in Eq. (2.21) [83].

The above analysis considers a minutiae correspondence based solely on the minutiae location. Next we consider a minutiae correspondence that depends on minutiae directions as well as minutiae positions. For the sake of this analysis, let us assume that the minutiae directions are completely independent of the minutiae positions and matching minutiae position and minutiae direction are therefore independent events.

Let $l$ be such that $P\left(\min\left(\left|\theta'_i - \theta_j\right|, 360 - \left|\theta'_i - \theta_j\right|\right) \leq \theta_0\right) = \frac{1}{l}$ in Eq. (2.14). Given $n$ input and $m$ template minutiae, the probability of $\rho$ minutiae falling into the *similar* positions can be estimated by Eq. (2.21). Once $\rho$ minutiae positions are matched, the probability that $q$ ($q \leq \rho$) minutiae among them have similar directions is given by

$$\binom{\rho}{q} \left(\frac{1}{l}\right)^q \left(\frac{l-1}{l}\right)^{\rho-q}, \tag{2.22}$$

where $\frac{1}{l}$ is the probability of two position-matched minutiae having a similar direction and $\frac{l-1}{l}$ is the probability of two position-matched minutiae taking different directions. Therefore, probability of matching $q$ minutiae in both position as well as direction is given by

$$p(M, m, n, q) = \sum_{\rho=q}^{\min(m,n)} \left( \frac{\binom{m}{\rho}\binom{M-m}{n-\rho}}{\binom{M}{n}} \times \binom{\rho}{q} \left(\frac{1}{l}\right)^q \left(\frac{l-1}{l}\right)^{\rho-q} \right) \tag{2.23}$$

Until now, we have assumed that the minutiae locations are uniformly distributed within the *entire* fingerprint area. Since $A$ is the area of the overlap between the template and the input fingerprints, the ridges occupy approximately $\frac{A}{2}$ of the area, with the other half being occupied by the valleys. Since the minutiae can lie only on ridges, i.e., along a curve of length $\frac{A}{w}$, where $w$ is the ridge period, the value of $M$ in Eq. (2.23) should therefore be changed from $M = A/C$ to $M = \frac{A/w}{2r_0}$, where $2r_0$ is

the length tolerance in minutiae location.

### Parameter Estimation

Our individuality model has several parameters, namely, $r_0$, $l$, $w$, $A$, $m$, $n$, and $q$. The value of $l$ further depends on $\theta_0$. The values of $r_0$, $l$, and $w$ are estimated in this section for a given sensor resolution. To compare the values obtained from the theoretical model with the empirical results, we will estimate the values of $A$, $m$, and $n$ from two different databases in the next section.

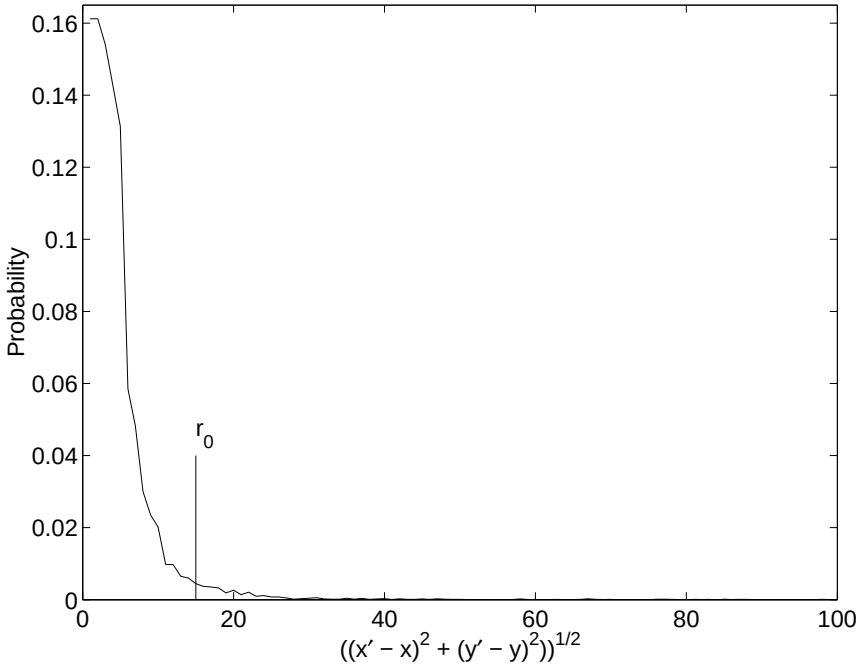
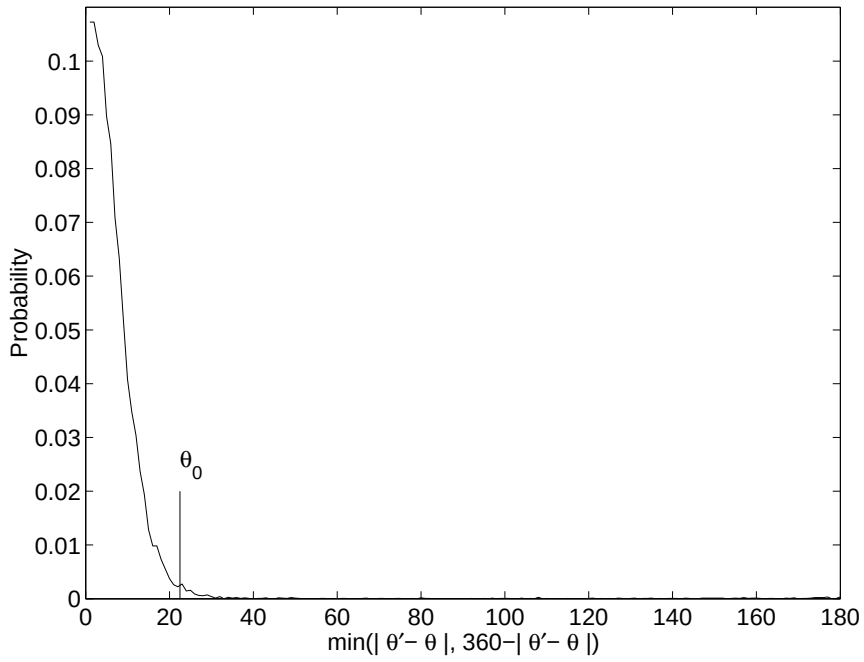



Figure 2.11: Distribution of minutiae distance differences for the genuine fingerprint pairs in the $GT$ database.

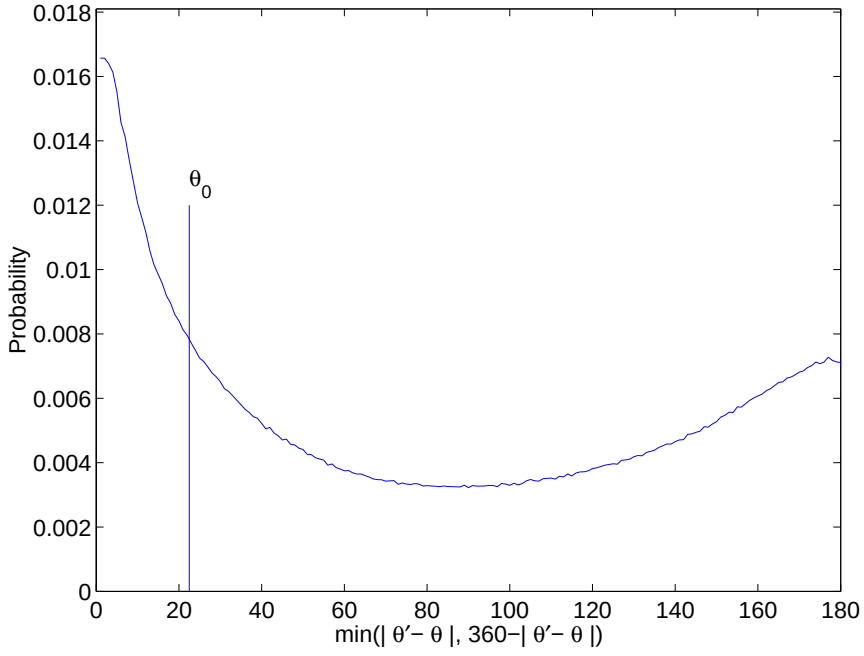
The value of $r_0$ should be determined to account for the variation in the different impressions of the same finger. However, since the spatial tolerance is dependent upon the scale at which the fingerprint images are scanned, we need to calculate it for the specific sensor resolution. We used a database (called $GT$) consisting of

450 mated pairs of fingerprints acquired using a high quality (Identicator [81]) optical scanner at a resolution of 500 *dpi*. The second print in the mated pair was acquired at least a week after the first print. The minutia were manually extracted from the prints by a fingerprint expert. The expert also determined the correspondence information for the detected minutiae. Using the ground truth correspondence information between duplex (two) pairs of corresponding minutiae, a rigid transformation between the mated pair was determined. The overall rigid transformation between the mated pair was determined using a least square approximation of the candidate rigid transformations estimated from each duplex pairs of the corresponding minutiae. After aligning a given mated pair of fingerprints using the overall transformation, the location difference $(x'-x, y'-y)$ for each corresponding minutia pair was computed; distance $(\sqrt{(x'-x)^2+(y'-y)^2})$ estimates for all minutiae in all mated fingerprint pairs were pooled to obtain a distribution for the distance between the corresponding minutiae (see Figure 2.11). We are seeking that value of $r_0$ for which $P\left(\sqrt{(x'-x)^2+(y'-y)^2} \leq r_0\right) \geq 0.975$, i.e., the value of $r_0$ which accounts for at least 97.5% of variation in the minutiae position of genuine fingerprint matchings. Thus, $r_0$ is determined from the distribution of $\sqrt{(x'-x)^2+(y'-y)^2}$ estimated in Figure 2.11 and is found to be 15 pixels for fingerprint images scanned at 500 *dpi* resolution.

To estimate the value of $l$, we first estimate the value of $\theta_0$. The value of $\theta_0$ can also be estimated using the database $GT$. After aligning a given mated pair of fingerprints using the overall transformation, we seek that value of $\theta_0$ which accounts for 97.5% variation in the minutia angles in the genuine fingerprint matchings, i.e.,

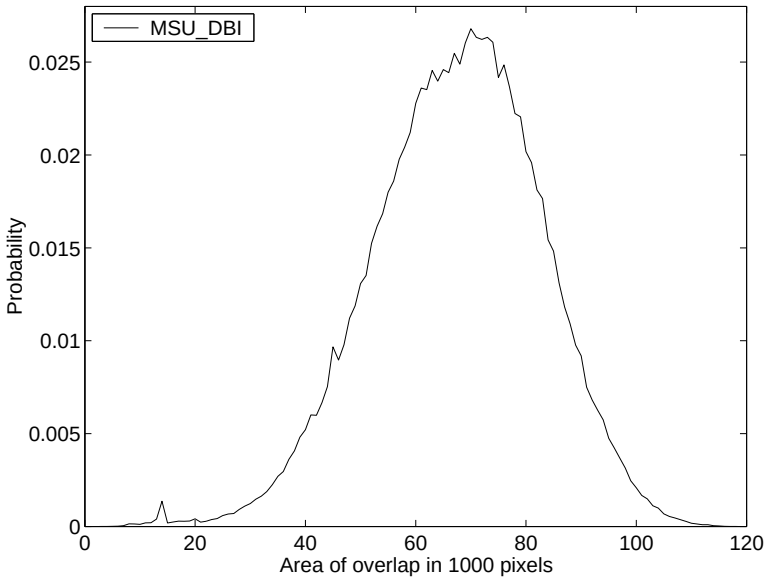



(a)

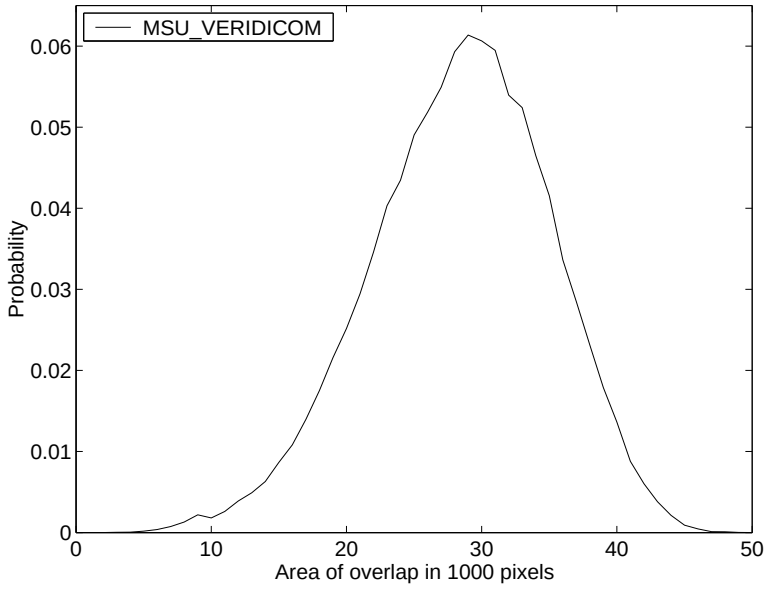



(b)

Figure 2.12: Distributions for minutiae angle differences for the (a) genuine fingerprint pairs using the ground truth and (b) imposter matchings using the automatic fingerprint matching system.




(a)




(b)

Figure 2.13: Area of overlap between the two fingerprints that are matched based on the bounding boxes of the minutiae features for (a) MSU_DBI database; (b) MSU_VERIDICOM database.

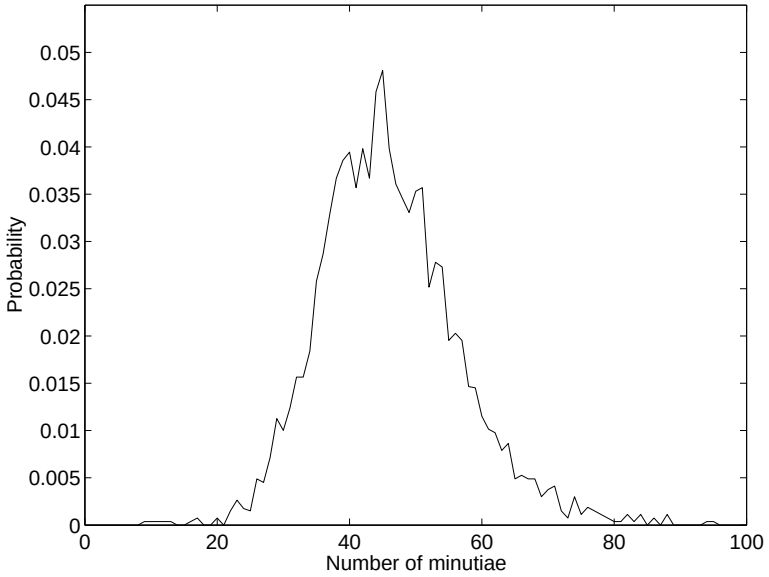
we seek that value of $\theta_0$ for which $P\left(\min\left(|\theta'_i - \theta_j|, 360 - |\theta'_i - \theta_j|\right) \leq \theta_0\right) \geq 0.975$. The distribution, $P\left(\min\left(|\theta' - \theta|, 360 - |\theta'_i - \theta_j|\right)\right)$ for the genuine fingerprint matchings in $GT$ is shown in Figure 2.12(a). Note that the minimum of the distribution occurs at 90° and the distribution between 90° and 180° is monotonically increasing. The area under this density from 90° to 180° is about 0.5% of the total area and quantifies the "connective ambiguity" (transformation of a ridge ending and a ridge bifurcation and *v*ice versa due to finger pressure variations). We believe that since the connective ambiguity is small (about 0.5%), it could be ignored. The value for $\theta_0$ for which $P\left(\min\left(|\theta' - \theta|, 360 - |\theta' - \theta|\right) \leq \theta_0\right) \geq 0.975$ is found to be $\theta_0 = 22.5°$. In the second step, we determine the distribution $P\left(\min\left(|\theta' - \theta|, 360 - |\theta' - \theta|\right)\right)$ for the imposter fingerprint matchings. Since we do not have correspondences marked by an expert between imposter fingerprint pairs, we depend on our fingerprint matcher to establish correspondences between minutiae in imposter pairs. Thus, our estimation of $l$ is slightly dependent on the automatic fingerprint matcher used but we believe that the value estimated here is very close to the true value of $l$. The distribution $P\left(\min\left(|\theta'_i - \theta_j|, 360 - |\theta'_i - \theta_j|\right)\right)$ estimated by using our matcher on the $GT$ database is shown in Figure 2.12(b) from which we determined that $P\left(\min\left(|\theta'_i - \theta_j|, 360 - |\theta'_i - \theta_j|\right) \leq 22.5°\right) = 0.267$, i.e., $l = 3.75$. Note that under the assumption that minutiae directions are uniformly distributed and the minutiae directions for the minutiae that match in their location are independent, we obtain $l = \frac{360}{2 \times 22.5} = 8$. If minutiae orientations (0-180) are considered instead of directions (0-360) , the value for $l$ determined from the experiments is 2.4 as opposed to a value of 4 determined under the assumption stated above.

The value of $w$ was taken as reported by Stoney [48]. Stoney estimated the value of ridge period as 0.463 mm/ridge from a database of 412 fingerprints. For fingerprint sensors with a resolution of 500 *dpi*, the ridge period converts to $\sim 9.1$ pixels/ridge. Thus, $w \sim 9.1$.

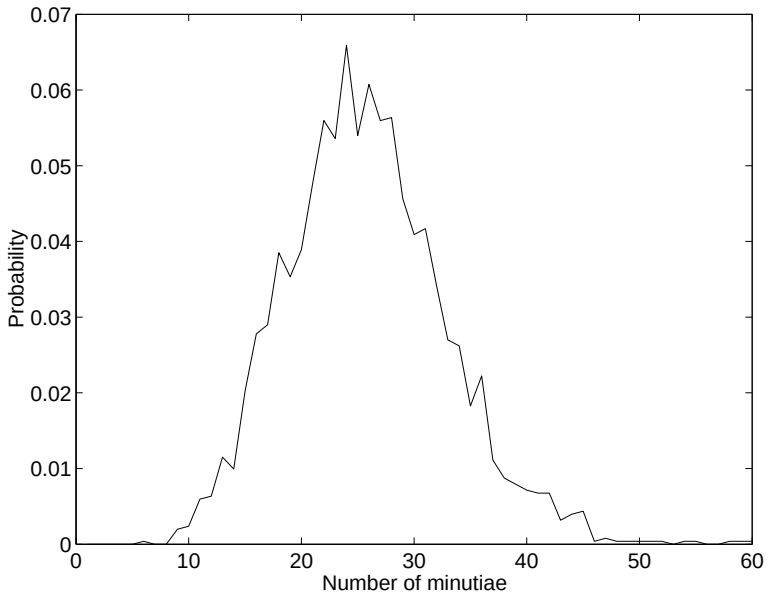
### 2.2.4 Experimental Results and Discussions

Fingerprint images were collected in our laboratory from 167 subjects using an optical sensor manufactured by Digital Biometrics, Inc. (image size = $508 \times 480$, resolution = 500 *dpi*). Single impressions of the right index, right middle, left index, and left middle fingers for each subject were taken in that order. This process was then repeated to acquire a second impression. The fingerprint images were collected again from the same subjects after an interval of six weeks in a similar fashion. Thus, we have four impressions for each of the four fingers of a subject. This resulted in a total of $2,672$ $(167 \times 4 \times 4)$ fingerprint images. We call this database MSU_DBI. A live feedback of the acquired image was provided and the subjects were guided in placing their fingers in the center of the sensor in an upright orientation. Using the protocol described above, we also collected fingerprint images using a solid-state fingerprint sensor manufactured by Veridicom, Inc. (image size = $300 \times 300$, resolution = 500 *dpi*). We call this database MSU_VERIDICOM. A large number of impostor matchings (over $4,000,000$) were generated using the automatic fingerprint matching system [11].

The mean values of $m$ and $n$ for impostor matchings were estimated as 46 for the

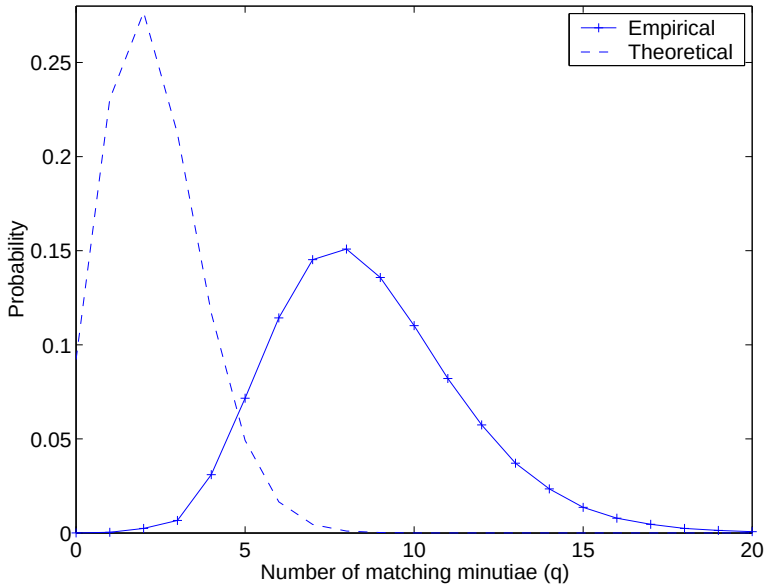



(a)

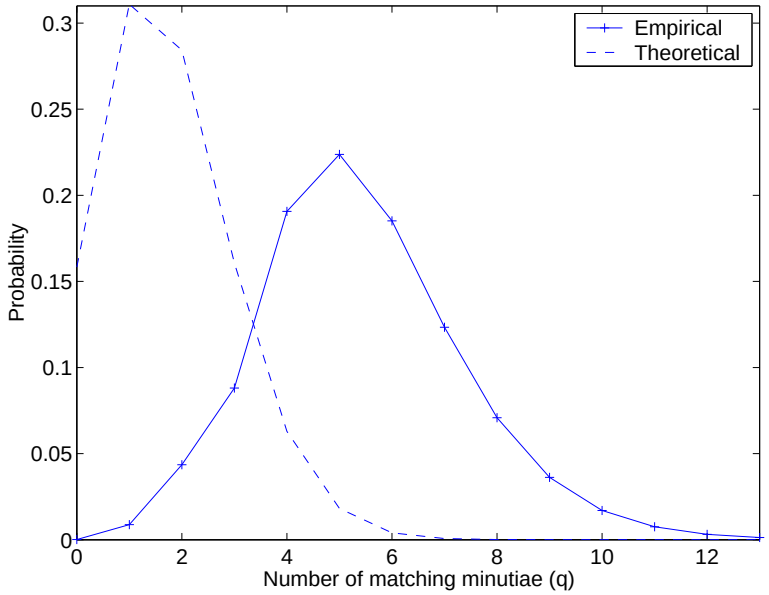



(b)

Figure 2.14: Distributions for $m$, $n$, and $q$ for computation of averages for (a) MSU_DBI database; (b) MSU_VERIDICOM database.




(a)




(b)

Figure 2.15: Comparison of experimental and theoretical probabilities for the number of matching minutiae. (a) MSU_DBI database; (b) MSU_VERIDICOM database.

MSU_DBI database and as 26 for the MSU_VERIDICOM database from the distributions of $m, n$ (Figures 2.14(a) and (b)). The average value of $A$ for the MSU_DBI and the MSU_VERIDICOM databases are $67,415$ pixels and $28,383$ pixels, respectively. The value of the overlapping area $A$ was estimated in the following fashion. After the template and the input fingerprints were aligned using the estimated transformation, a bounding box $A_i$ of all the corresponding minutiae in the input fingerprint was computed in the common coordinate system. Similarly, a bounding box $A_t$ of all the corresponding minutiae in the template fingerprint was also computed in the common coordinate system. The intersection $A$ of these two bounding boxes $A_i$ and $A_t$ for each matching was then estimated. The estimates of $A$ for all the matchings performed in the database were pooled to obtain a distribution for $A$ (see Figures 2.13 (a) and (b)). An arithmetic mean of the distribution was used to arrive at an estimate of $A$.

The probabilities of a fingerprint correspondence obtained for different values of $M$, $m$, $n$, and $q$ are given in Table 2.5. The values obtained from our model shown in Table 2.5 can be compared with values obtained from the previous models in Table 2.3 for $m = 36$, $n = 36$, and $q = 36, 12$.

Typically, a match consisting of 12 minutiae points (*the 12-point rule*) is considered as sufficient evidence in many courts of law. Assuming that an expert can correctly glean all the minutia in the latent, a 12-point match with the full-print template (see the first row, last column entry in Table 2.4) is an overwhelming amount of evidence, *provided* that there is no contradictory minutia evidence in the overlapping area. The value of $A$ was computed for 500 *dpi* fingerprint images from the minutiae

Table 2.4: The effects of the fingerprint expert misjudgments in using the 12-point rule. The source of error could be in underestimating the minutiae detected in the latent print ($n$) or overestimating the correct number of matched minutiae ($q$). $m = 12$ for all entries. Except for $(m = 12, n = 12, q = 12)$ entry, all other entries represent incorrect judgments by the fingerprint expert. For instance, the entry $(m = 12, n = 14, q = 8)$ in the table indicates that although the fingerprint examiner determined that 12 template minutia unequivocally matched with all 12 input minutiae, there were indeed 14 input minutiae (2 missed input minutiae) out of which only 8 correctly matched with the corresponding template minutiae (4 incorrect match judgments).

| $q$ / $n$ | 8 | 9 | 10 | 11 | **12** |
|---|---|---|---|---|---|
| **12** | $6.19 \times 10^{-10}$ | $4.88 \times 10^{-12}$ | $1.96 \times 10^{-14}$ | $3.21 \times 10^{-17}$ | $\mathbf{1.22 \times 10^{-20}}$ |
| 13 | $1.58 \times 10^{-9}$ | $1.56 \times 10^{-11}$ | $8.42 \times 10^{-14}$ | $2.08 \times 10^{-16}$ | $1.58 \times 10^{-19}$ |
| 14 | $3.62 \times 10^{-9}$ | $4.32 \times 10^{-11}$ | $2.92 \times 10^{-13}$ | $9.66 \times 10^{-16}$ | $1.11 \times 10^{-18}$ |
| 15 | $7.63 \times 10^{-9}$ | $1.06 \times 10^{-10}$ | $8.68 \times 10^{-13}$ | $3.60 \times 10^{-15}$ | $5.53 \times 10^{-18}$ |
| 16 | $1.50 \times 10^{-8}$ | $2.40 \times 10^{-10}$ | $2.30 \times 10^{-12}$ | $1.45 \times 10^{-14}$ | $2.21 \times 10^{-17}$ |

density of 0.246 minutiae/mm$^2$ estimated by Kingston (cf. [48]) from 100 fingerprints; thus $M = 35$. Since latents are typically of very poor quality, it is possible that there could be an error in judgment of existence of minutiae in the latent or their possible match to the minutiae in the template print. The effect of such misjudgments on the chance of false associations is rather dramatic. For instance, imposing two incorrect minutiae match judgments lowers the probability of the match from $1.22 \times 10^{-20}$ to $1.96 \times 10^{-14}$ and ignoring two genuine minutiae present in the input (latent) print lowers the probability from $1.22 \times 10^{-20}$ to $1.11 \times 10^{-18}$. Thus, the misjudgment of a false minutiae match has significantly more impact that that of missing genuine minutiae in the input latent print.

Figures 2.15(a) and (b) show the distribution of the number of matching minutiae computed from the MSU_DBI and MSU_VERIDICOM databases using an automatic fingerprint matching system (AFMS) [11], respectively. These figures also show the

Table 2.5: Fingerprint correspondence probabilities obtained from the proposed individuality model for different sizes of fingerprint images containing 26, 36 or 46 minutiae. $M$ for the last entry was computed by estimating typical print area manifesting 12 minutia in a 500 dpi optical fingerprint scan. The entry $(35, 12, 12, 12)$ corresponds to the 12-point rule.

| $M$, m, n, q | P(Fingerprint Correspondence) |
|---|---|
| 104, 26, 26, 26 | $5.27 \times 10^{-40}$ |
| 104, 26, 26, 12 | $3.87 \times 10^{-9}$ |
| 176, 36, 36, 36 | $5.47 \times 10^{-59}$ |
| 176, 36, 36, 12 | $6.10 \times 10^{-8}$ |
| 248, 46, 46, 46 | $1.33 \times 10^{-77}$ |
| 248, 46, 46, 12 | $5.86 \times 10^{-7}$ |
| 70, 12, 12, 12 | $1.22 \times 10^{-20}$ |

theoretical distributions obtained from our model described in Section 2.2.3 for the average values of $M$, $m$, and $n$ computed from the databases. The empirical distribution is to the right of the theoretical distribution, which can be explained by the following factors: $(i)$ some true minutiae are missed and some spurious minutiae are detected by the automatic system due to noise in the fingerprint images and the imperfect nature of the automatic algorithms. Spurious minutiae may also be detected because of cuts and bruises on the fingertips; $(ii)$ the automatic matching algorithm cannot completely recover the non-linear deformation present in the fingerprint images; so the alignment between the input and template has some error. $(iii)$ automatic feature extraction introduces error in minutiae location and orientations. $(iv)$ the matcher seeks that alignment which maximizes the number of minutiae correspondences. Consequently, the chance of false associations increases.

The theoretical curve is the upper bound on the performance of a minutiae-based automatic fingerprint verification system which means that it is possible to improve the system to match the theoretical curve. At the same time, the automatic sys-

tem can not perform better than the theoretical limit because of limited information content in the minutiae-based matching.

Table 2.6 shows the empirical probability of matching 10 and 15 minutiae in MSU_VERIDICOM and MSU_DBI databases, respectively. The typical values of $m$ and $n$ were estimated from their distributions by computing the arithmetic means. The probabilities of false correspondence for these values of $m$, $n$ and $q$, are reported in the third column of Table 2.6. Admittedly, this is an approximate procedure but we do not expect significant deviations from our probability estimates even when the exact procedure for estimating the probability is adopted.

Table 2.6: Fingerprint correspondence probabilities obtained from matching imposter fingerprints using an AFMS [11] for the MSU_VERIDICOM and MSU_DBI databases. The probabilities given in the table are for matching "exactly $q$" minutiae. The probabilities for matching "$q$ or more" minutiae are $3.0 \times 10^{-2}$ and $3.2 \times 10^{-2}$ for the MSU_VERIDICOM and MSU_DBI databases, respectively, i.e., of the same order. The average values for $M$, $m$, and $n$ are 28, 383, 26, and 26 for the MSU_VERIDICOM database and 67, 415, 46 and 46 for the MSU_DBI database, respectively.

| Database | m,n,q | P(False Correspondence) |
|---|---|---|
| MSU_VERIDICOM | 26, 26, 10 | $1.7 \times 10^{-2}$ |
| MSU_DBI | 46, 46, 15 | $1.4 \times 10^{-2}$ |

### 2.2.5 Summary

One of the most fundamental questions one would like to ask about any *practical* biometric authentication system is: what is the inherent discriminable information available in the input signal? Unfortunately, this question, if at all, has been answered in a very limited setting for most biometrics modalities, including fingerprints. The inherent signal capacity issue is of enormous complexity as it involves modeling both

the composition of the population as well as the interaction between the behavioral and physiological attributes at different scales of time and space. Nevertheless, a first-order approximation to the answers to these questions will have a significant bearing on the acceptance of fingerprint- (biometrics-) based personal identification systems into our society as well as determining the upper bounds on scalability of deployments of such systems.

Estimating fingerprint individuality essentially involves determining the discriminatory information within the input measurements (fingerprint images) to resolve the identities of the people. The empirical and theoretical methods of estimating individuality serve complementary goals. Empirical observations lead us to characterize the constraints on the discriminatory information across different fingers as well as the invariant information among the different impressions of the same finger; the theoretical modeling/generalization of these constraints permits a prediction of the bounds on the performance and facilitates development of constructive methods for an independent empirical validation. Historically, there has been a disconnect in the performance evaluations of practical fingerprint systems and theoretical performance predictions. Further, the data-dependent empirical performance evaluations themselves have varied quite dramatically.

The model proposed here is relatively simple. It ignores most of the known (weak) dependencies among the features and does not directly include features such as ridge counts, fingerprint class, ridge frequencies, permanent scars, etc. For these reasons, we suspect that the proposed model does not yet compete in predicting the performance of human fingerprint expert matcher. Yet, we believe that the individuality estimates

predicted by the present model are significantly closer to the performance of practical automatic fingerprint matchers on realistic data samples than other models reported in the literature.

While the individuality of the minutiae based fingerprint representation based on our model is lower than the previous estimates, our study indicates that the likelihood of an adversary guessing someone's fingerprint pattern (e.g., requiring matching 20 or more minutia from a total of 36) is significantly lower than a hacker being able to guess a six-character alpha-numerical case-sensitive (most probably weak) password by social engineering techniques (most common passwords are based on birthday, spouse's name, etc.) or by brute force. Obviously, more stringent conditions on matching will provide a better cryptographic strength at the risk of increasing the false negative error rate.

If a typical full dab fingerprint contains 46 minutiae, there is an overwhelming amount of information present in the minutiae representation of fingerprints for manual identification (the probability of a false correspondence between two fingerprints from different users containing 46 minutiae each is $1.33 \times 10^{-77}$). However, an automatic system that makes its decision based on 12 minutiae correspondences is utilizing only a limited amount of information (the probability of a false correspondence for matching 12 minutiae between two fingerprints from different users containing 46 minutiae each is $5.86 \times 10^{-7}$). Due to this limited amount of information present in the minutiae representation of fingerprints, it is desirable to explore alternate complementary representations of fingerprints for automatic matching. In Chapter 3, we describe such an alternate texture-based representation of fingerprints and empirically

show that it has a discriminatory power similar to the minutiae-based representation.

# Chapter 3

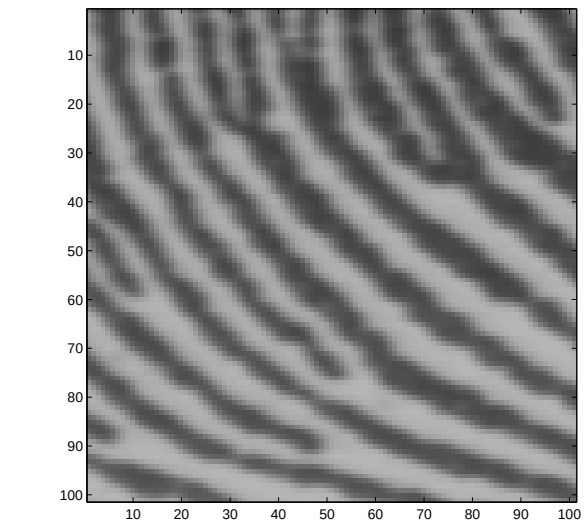
# Fingerprint as Oriented Texture

Traditionally, there are two main types of features in fingerprints: $(i)$ global ridge and furrow structures which form a special pattern in the central region of the fingerprints, and $(ii)$ minute details associated with local ridges and furrows. A fingerprint is typically classified based on only the first type of features and uniquely identified based on the second type of features. The minutiae-based representation is the most popular representation of fingerprints as it has a long history of use by the forensic experts who visually match fingerprints. Forensic experts also use other features such as ridge count between pairs of minutiae and ridge width in conjunction with minutiae for identification purposes. However, automatic processing of fingerprints allows the use of Cartesian coordinates and Euclidean distances in establishing the similarity between fingerprints. Similarly, the use of an alternate representation of fingerprint that has good discriminatory power is also feasible for automatic systems. Chapter 2 has established an upper bound on the performance of minutiae-based fingerprint matching systems due to the limited amount of information content present in the

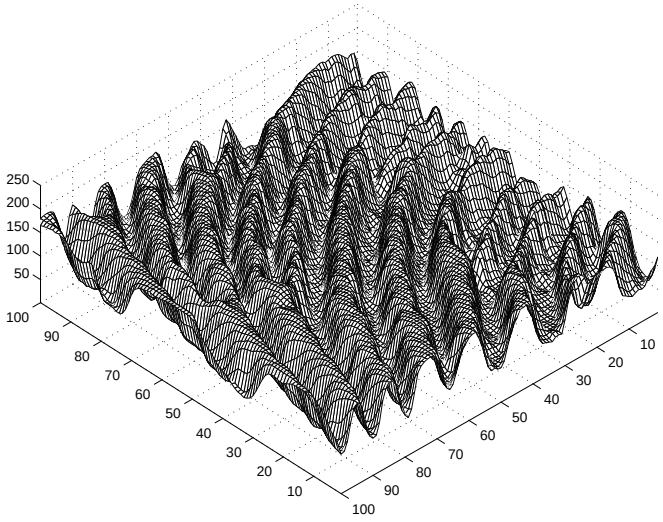
minutiae-based representation. As a result, it is desirable to explore an alternate independent representation of fingerprints that can complement the minutiae-based representation. This complementary representation should combine both the global and the local information sources in a fingerprint to obtain a rich representation. This representation should not only take into account the local anomalies in the ridge structure (e.g., minutiae), but also, for instance, the global pattern of ridges and furrows, inter-ridge distances, and overall patterns of ridge flow. Further, it is an added advantage to design representations which can be automatically and reliably extracted from the fingerprint and whose extraction will degrade gracefully with deterioration in the quality of the fingerprints.

## 3.1 Introduction

The smooth flow pattern of ridges and valleys in a fingerprint can be viewed as an oriented texture field [28] (see Figure 3.1). The image intensity surface in fingerprint images is comprised of ridges whose directions vary continuously, which constitutes an oriented texture. Most textured images contain a limited range of spatial frequencies, and mutually distinct textures differ significantly in their dominant frequencies [2, 84, 10]. Textured regions possessing different spatial frequency, orientation, or phase can be easily discriminated by decomposing the image in several spatial frequency and orientation channels. For typical fingerprint images scanned at 500 *dpi*, there is very little variation in the spatial frequencies (determined by inter-ridge distances) among different fingerprints. This implies that there is an optimal scale (spatial frequency)




(a)




(b)

Figure 3.1: Flow pattern in a fingerprint image. (a) A section of a fingerprint image, (b) 3-dimensional surface plot of (a).

for analyzing the fingerprint texture. Every pixel in a fingerprint image is associated with a dominant local orientation and a local measure of coherence of the flow pattern. A symbolic description of a fingerprint image can be derived by computing the angle and coherence at each pixel in the image. Fingerprints can be represented/matched by using quantitative measures associated with the flow pattern (oriented texture) as features.

Analysis and modeling of oriented textures is an important research problem with a wide variety of practical applications [28]. Previous attempts at describing oriented textures have used either exclusively local or predominantly global features. Examples of local representations include Poincaré indices, winding numbers, and information related to singularities and anomalies. The primary limitation of the local approaches to representation of an oriented texture is that it does not efficiently capture the gross discriminatory information. Local information also tends to be unstable and noise prone. Examples of global representations include directional co-occurrence matrices, phase portraits of the orientation fields, and autocorrelation methods. Jain and Farrokhnia [10] derived a global representation of texture by decomposing the input image into different frequency and orientation components using a Gabor filterbank. They applied this representation to successfully classify and segment textured images. The global representations, although efficient, do not capture all the discriminatory information. For example, the global configuration of the two fingerprints shown in Figure 3.2 is the same but the prints are different due to different configuration of the local anomalies. Discriminating individual members of such a texture family based on global representations alone is not feasible.

(a) (b)

Figure 3.2: Difficulty in fingerprint matching. (a) and (b) have the same global configuration but are images of two different fingers.

Daugman [86] derived a translation and scale invariant texture representation (called *IrisCode*) for the human iris by an ordered enumeration of multi-scale quadrature Gabor wavelet coefficients of the visible iris texture. Daugman's iris texture representation is not rotation invariant. But large rotations in human iris do not occur due to the restricted movement of the head. Small amounts of rotation were handled in the the matching phase by a rotation of the IrisCode itself. Our representation for oriented texture of fingerprints was inspired by Daugman's work on iris recognition and the success of the Gabor filterbank as reported by Jain and Farrokhnia [10] . We propose a generic scheme for representing fingerprint texture that relies on extracting one (or more) invariant points of reference of the fingerprint texture based on an analysis of its orientation field. A predetermined region of interest around the reference point is tessellated into cells. Each cell is then examined for the information in one

or more different, orientation specific, spatial frequency channels. An ordered enumeration of the features thus extracted from each cell is used as the representation of the fingerprint (see Figure 3.3). Thus, the representation elements capture the local information and the ordered enumeration of the tessellation captures the invariant global relationships among the local patterns.

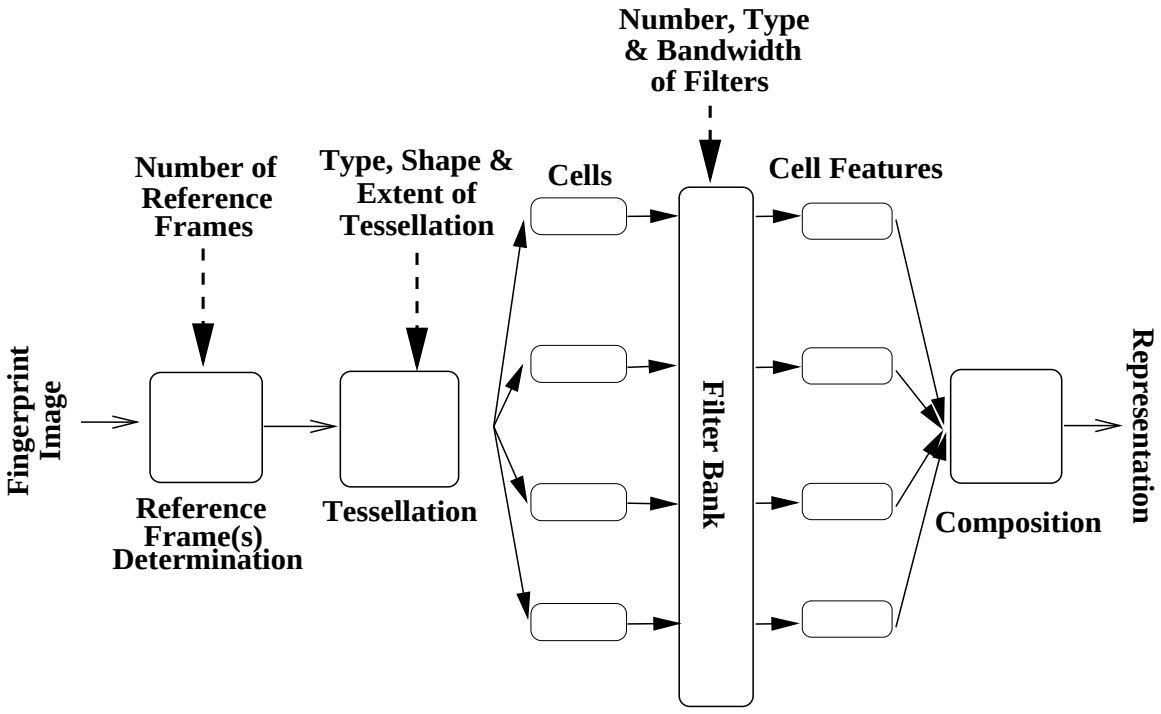



Figure 3.3: Schematic diagram for extraction of generic texture-based representation for fingerprints.

It is desirable to obtain representations for fingerprints which are scale (due to pressure and sensor resolution), translation, and rotation invariant. Scale invariance is not a significant problem since most fingerprint images could be scaled as per the *dpi* specification of the sensors. Figure 3.4 shows that a child's fingerprint has a smaller area when both the fingerprints were scanned at the same *dpi* resolution. When a

(a)

(b)

Figure 3.4: Fingerprint of (a) a child, and (b) an adult. Both the fingerprints were scanned at 500 *dpi*.

child grows up, the scale difference between his fingerprints acquired at different ages may result in a fingerprint mismatch. Periodically updating the fingerprint template will alleviate this problem. The translation invariance is accomplished by locating the reference point. The representation proposed here is not rotation invariant and so the rotation is handled by a rotation of the representation in the matching stage. A circular tessellation is defined so that a rotation in the fingerprint image corresponds to a cyclic rotation of the elements of the representation. The local discriminatory information in the sector needs to be decomposed into separate components. A Gabor filterbank is one of the well-known techniques to capture useful information in specific bandpass channels as well as to decompose this information into orthogonal components in terms of spatial frequencies. The four main steps in our representation extraction algorithm are: (*i*) determine a reference point for the fingerprint image, (*ii*) tessellate the region around the reference point, (*iii*) filter the region of interest in eight different directions using a bank of Gabor filters (eight directions are required to completely capture the local ridge characteristics in a fingerprint while only four directions are required to capture the global configuration [18]), and (*iv*) compute the average absolute deviation from the mean (AAD) of gray values in individual sectors in filtered images to define the feature vector, also called the FingerCode (similar to the IrisCode introduced by Daugman [86]).

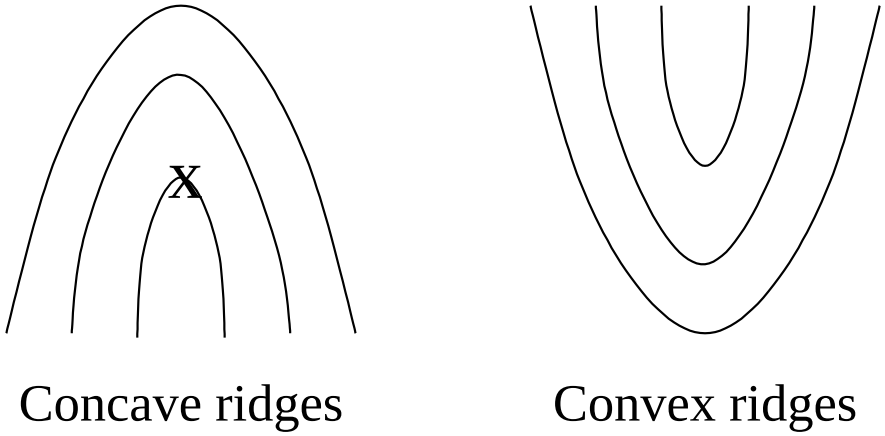



Figure 3.5: Concave and convex ridges in a fingerprint image when the finger is positioned upright. The reference point is marked by $X$.

## 3.2 Reference Point Location

Fingerprints have many conspicuous landmarks and any combination of them could be used for establishing a reference point. We define the reference point of a fingerprint as the point of maximum curvature of the concave ridges (see Figure 3.5) in the fingerprint image.

Many previous approaches to determination of a reference point $(x_c, y_c)$ critically relied on the local features like Poincaré index [97] or some other similar properties of the orientation field. While these methods work well for good quality fingerprint images, they fail to correctly localize reference points in poor quality fingerprints with cracks and scars, dry skin, or poor ridge and valley contrast. Recently, Hong and Jain [105] have attempted to judiciously combine the orientation field information with available ridge details for fingerprint classification. However, their method does not reliably handle poor quality fingerprints when the orientation field is very noisy and it can be misled by poor structural cues in the presence of finger cuts and bruises

on the skin.

In order that a reference point algorithm gracefully handle local noise in a poor quality fingerprint, the detection should necessarily consider a large neighborhood in the fingerprint image. On the other hand, for an accurate localization of the reference point, the approach should be sensitive to the local variations in a small neighborhood. To meet these conflicting requirements of an accurate and reliable localization, we propose a new method of reference point determination based on multi-resolution analysis of the orientation fields. This method locates the reference point more precisely than the algorithm proposed by Hong and Jain [105].

Given an $M \times N$ fingerprint image, $I$, its *orientation field*, $\mathcal{O}$, is defined as an $P \times Q$ image, where $\mathcal{O}(i,j)$ represents the local ridge orientation at pixel $(i,j)$, $P \leq M, Q \leq N$. Local ridge orientation is usually specified for a block rather than at every pixel in the image $I$. The fingerprint image is divided into a set of $w \times w$ non-overlapping blocks and a single orientation is defined for each block (see Figures 3.6 (a) and (b)); $P = \lfloor \frac{M}{w} \rfloor, Q = \lfloor \frac{N}{w} \rfloor$. Note that there is an ambiguity by a factor of $\pi$ in fingerprint orientation, i.e., local ridges oriented at $\frac{\pi}{2}$ and ridges oriented at $\frac{3\pi}{2}$ cannot be differentiated from each other. A number of methods have been developed to estimate the orientation field in a fingerprint [119, 162, 120, 28]. The least mean square orientation estimation algorithm [108] used here has the following steps:

1. Divide $\mathcal{I}$, the input fingerprint image, into non-overlapping blocks of size $w \times w$.

2. Compute the gradients $\partial_x(i,j)$ and $\partial_y(i,j)$ at each pixel $(i,j)$. Depending on the computational requirement, the gradient operator may vary from the simple

*Sobel* operator to the more complex *Marr-Hildreth* operator [59].

3. Estimate the local orientation of each block centered at pixel $(i, j)$ using the following equations [28]:

$$\mathcal{V}_x(i,j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} 2\partial_x(u,v)\partial_y(u,v), \tag{3.1}$$

$$\mathcal{V}_y(i,j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} (\partial_x^2(u,v) - \partial_y^2(u,v)), \tag{3.2}$$

$$\mathcal{O}(i,j) = \frac{1}{2} tan^{-1}(\frac{\mathcal{V}_y(i,j)}{\mathcal{V}_x(i,j)}), \tag{3.3}$$

where $\mathcal{O}(i, j)$ is the least square estimate of the local ridge orientation of the block centered at pixel $(i, j)$. Mathematically, it represents the direction that is orthogonal to the dominant direction of the *Fourier spectrum* of the $w \times w$ window.

A summary of our reference point location algorithm is presented below:

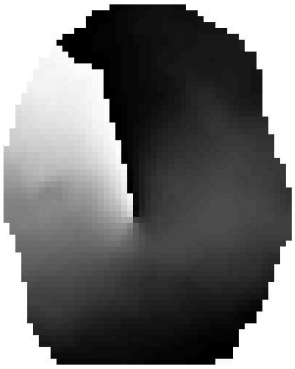
1. Estimate the orientation field $\mathcal{O}$ as described above using a window size of $w \times w$.

2. Smooth the orientation field in a local neighborhood. Let the smoothed orientation field be represented as $\mathcal{O}'$. In order to perform smoothing (low-pass filtering), the orientation image needs to be converted into a *continuous vector field*, which is defined as follows:

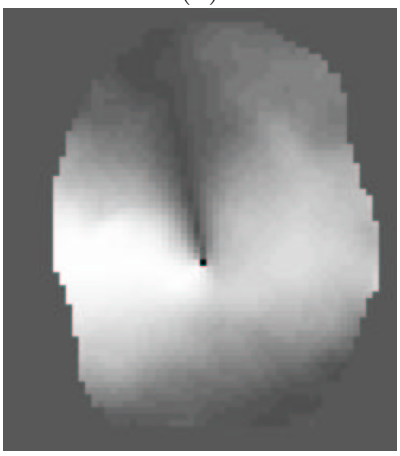
$$\Phi_x(i,j) = \cos(2\mathcal{O}(i,j)), \ \textit{and} \tag{3.4}$$

$$\Phi_y(i,j) = \sin(2\mathcal{O}(i,j)), \tag{3.5}$$

(a)



(b)



(c)

Figure 3.6: Estimating the reference point. (a) Smoothed orientation field overlapped on the original image, (b) orientation field ($w$=10) shown as intensity distribution; the background has been segmented, and (c) *sine* component of the orientation field; the darkest pixel in the center of the image marks the detected reference point. Images have been scaled to the range 0-255 for viewing.

where $\Phi_x$ and $\Phi_y$, are the $x$ and $y$ components of the vector field, respectively. A low-pass filtering of the resulting vector field is performed as follows:

$$\Phi'_x(i,j) = \sum_{u=-w_\Phi/2}^{w_\Phi/2} \sum_{v=-w_\Phi/2}^{w_\Phi/2} W(u,v)\Phi_x(i-uw, j-vw) \;\; and \tag{3.6}$$

$$\Phi'_y(i,j) = \sum_{u=-w_\Phi/2}^{w_\Phi/2} \sum_{v=-w_\Phi/2}^{w_\Phi/2} W(u,v)\Phi_y(i-uw, j-vw), \tag{3.7}$$

where $W$ is a $w_\Phi \times w_\Phi$ low-pass filter with unit integral. Note that the smoothing operation is performed at the block level. For our experiments, we used a $5 \times 5$ mean filter. The smoothed orientation field $\mathcal{O}'$ at $(i,j)$ is computed as follows

$$\mathcal{O}'(i,j) = \frac{1}{2}\tan^{-1}(\frac{\Phi'_y(i,j)}{\Phi'_x(i,j)}). \tag{3.8}$$

3. Compute $\mathcal{E}$, an image containing only the *sine* component of $\mathcal{O}'$.

$$\mathcal{E}(i,j) = sin\left(\mathcal{O}'(i,j)\right). \tag{3.9}$$

4. Initialize $\mathcal{A}$, a label image used to indicate the reference point.

5. For each pixel $(i,j)$ in $\mathcal{E}$, integrate pixel intensities (sine component of the orientation field) in regions $R_I$ and $R_{II}$ shown in Figure 3.7 and assign the corresponding pixels in $\mathcal{A}$ the value of their difference.

$$\mathcal{A}(i,j) = \sum_{R_I} \mathcal{E}(i,j) - \sum_{R_{II}} \mathcal{E}(i,j). \tag{3.10}$$

The regions $R_I$ and $R_{II}$ (see Figure 3.7) were determined empirically by applying the reference point location algorithm over a large database. The radius of the semi-circular region was set equal to the window size $w$. The geometry of regions $R_I$ and $R_{II}$ is designed to capture the maximum curvature in concave ridges (see Figure 3.5). Although this approach successfully detects the reference point in most of the cases, including double loops (see Figure 3.8 (a)), the present implementation is not very precise and consistent for the arch type fingerprints because it is difficult to localize points of high curvature in arch type fingerprint images.

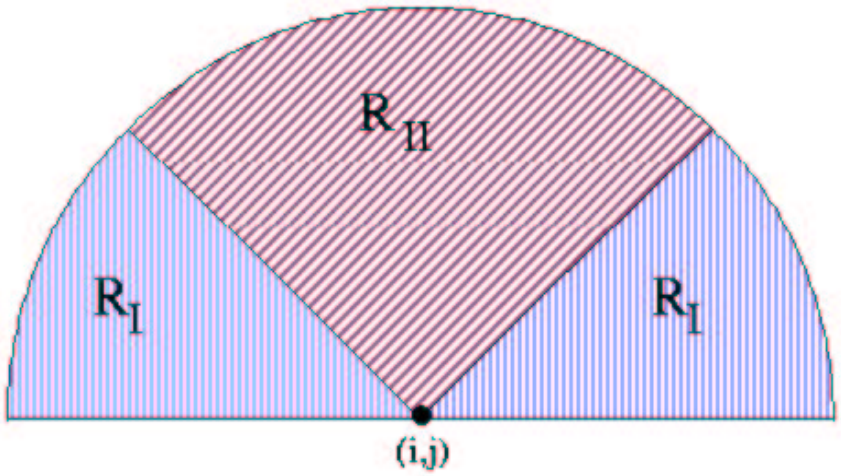



Figure 3.7: Regions for integrating pixel intensities in $\mathcal{E}$ for computing $\mathcal{A}(i, j)$.

6. Find the maximum value in $\mathcal{A}$ and assign its coordinate to the core, i.e., the reference point.

7. For a fixed number of times, repeat steps 1-6 by using a window size of $w' \times w'$, where $w' < w$ and restrict the search for the reference point in step 6 in a local neighborhood of the detected reference point. In our experiments, we used three

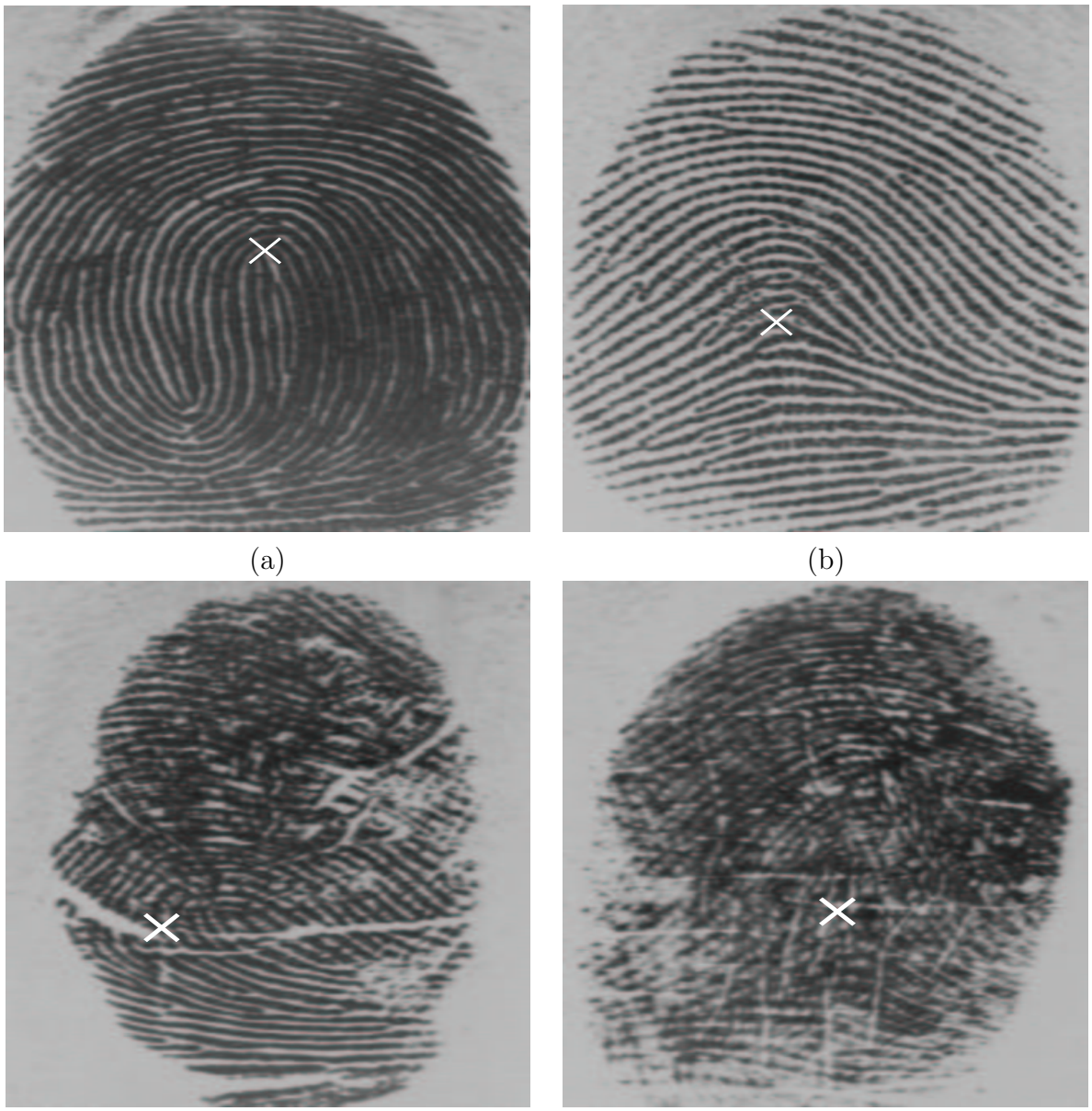
iterations with $w = 15, 10$, and 5 pixels respectively, and hence the precision of the detected reference point is 5 pixels.

Figure 3.8 shows the results of our reference point location algorithm for four different images. The reference point location algorithm performs extremely well for good quality fingerprint images of whorl, left loop, right loop, and arch types. This algorithm has higher error in consistently locating the reference point in the arch type fingerprints due to the absence of singular points in arch type fingerprint images. The algorithm fails for very poor quality fingerprints because of the errors in orientation field estimation.

## 3.3 Tessellation

Let $\mathcal{I}(x, y)$ denote the gray level at pixel $(x, y)$ in an $M \times N$ fingerprint image and let $(x_c, y_c)$ denote the reference point. The *region of interest* in the fingerprint is defined as the collection of all the sectors $S_i$, where the $i^{th}$ sector $S_i$ is computed in terms of parameters $(r, \theta)$ as follows:

$$\begin{aligned} S_i &= \{(x, y) \, | b(T_i + 1) \leq r < b(T_i + 2), \\ &\quad \theta_i \leq \theta < \theta_{i+1} \ , 1 \leq x \leq N, 1 \leq y \leq M\} \, , \end{aligned} \tag{3.11}$$



(a) (b)

(c) (d)

Figure 3.8: Examples of the results of our reference point location algorithm. The algorithm fails on very poor quality fingerprints such as (c) and (d).

where

$$T_i = i \; div \; k, \tag{3.12}$$

$$\theta_i = (i \; mod \; k) \times (2\pi/k), \tag{3.13}$$

$$r = \sqrt{(x - x_c)^2 + (y - y_c)^2}, \tag{3.14}$$

$$\theta = tan^{-1}((y - y_c)/(x - x_c)), \tag{3.15}$$

$b$ is the width of each band, $k$ is the number of sectors considered in each band, and $i = 0, ..., (B \times k - 1)$, where $B$ is the number of concentric bands considered around the reference point for feature extraction. The parameter $B$ depends on the area of the finger imaged. For example, at the same resolution of 500 *dpi*, a larger finger area will be captured in a $640 \times 480$ pixel image than in a $320 \times 320$ pixel image. Thus the parameter $B$ depends on the image size and the *dpi* resolution of the sensor. The width of the concentric bands is defined by the parameter $b$ and depends on the *dpi* resolution of the sensor. The width of the bands should capture one ridge and valley pair on an average. For fingerprint images scanned at 500 *dpi*, we choose $b = 20$. A band with a width of 20 pixels is necessary to capture a single minutia in a sector, allowing our low-level features to capture this local information. If the sector width is more than 20 pixels, then the local information is modulated by more global information. The innermost band (circle) is not used for feature extraction because the sectors in the region near the reference point contain very few pixels and, therefore, the feature extraction in this region is not very reliable. A circular tessellation is chosen because a rotation of the fingerprint will correspond to the rotation of the tessellation. The value

of $k$ controls the capture of the global versus the local information in a fingerprint and depends upon the application. For example, more global information is required by the fingerprint classification algorithm, and so, a lower $k$ value is chosen. On the other hand, the fingerprint verification application needs to capture more local information in the fingerprints and hence requires a higher value of $k$. The values for these parameters, $B$, $b$, and $k$ were determined empirically to obtain the best performance for the fingerprint classification and matching applications. Both the classification and the matching algorithms based on the FingerCode representation are able to handle small changes in these parameters without a significant degradation in performance. A large change in the parameter values is also handled gracefully with a decrease in performance proportional to the change in the parameter values. The value of $B$ should be set in such as way as to capture maximum ridge and valley details without rejecting a large number of fingerprint images. The value of $k$ should be chosen based on the tradeoff between local and global information required for a particular application, the value of $b$ should be chosen based on the *dpi* resolution of the sensor and the average inter-ridge distance in fingerprint images. Once the parameter values are chosen for an application, they remain constant.

## 3.4 Filtering

Fingerprints have local parallel ridges and valleys, and well-defined local frequency and orientation (see Figure 3.10). Properly tuned Gabor filters [86, 88] can remove noise, preserve the true ridge and valley structures, and provide information contained

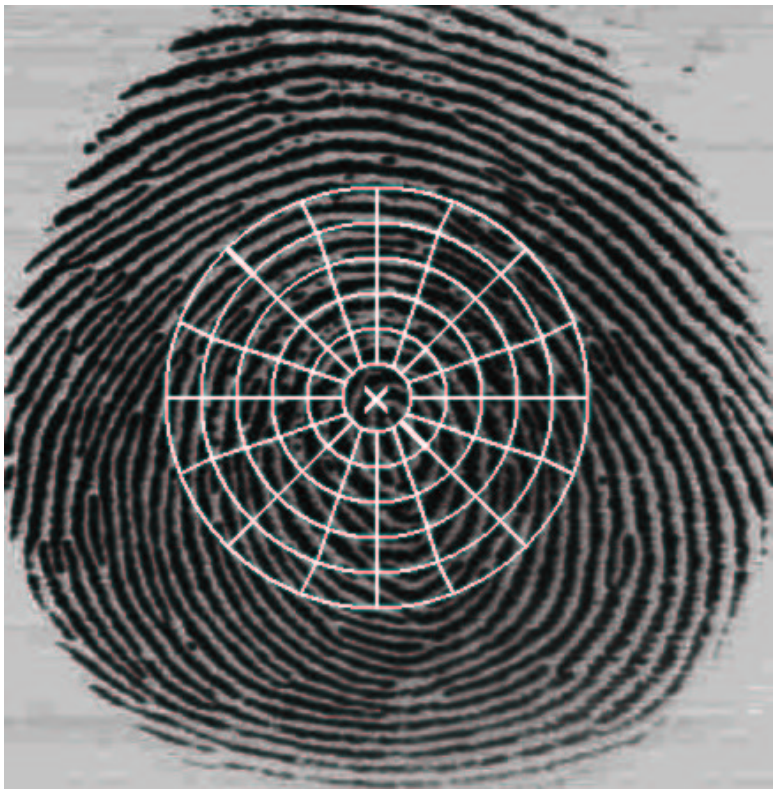


Figure 3.9: Reference point ($\times$), the region of interest, and 80 sectors ($B = 5$, $k = 16$) superimposed on a fingerprint.

in a particular orientation in the image. A minutia point can be viewed as an anomaly in locally parallel ridges and it is this information that we are attempting to capture using the Gabor filters.

Before filtering the fingerprint image, we normalize the grey level intensities in the region of interest in each sector separately to a constant mean and variance. Normalization is performed to remove the effects of sensor noise and gray level background due to finger pressure differences. Let $I(x, y)$ denote the gray value at pixel $(x, y)$, $M_i$ and $V_i$, the estimated mean and variance of grey levels in sector $S_i$, respectively, and $N_i(x, y)$, the normalized gray-level value at pixel $(x, y)$. For all the pixels in sector

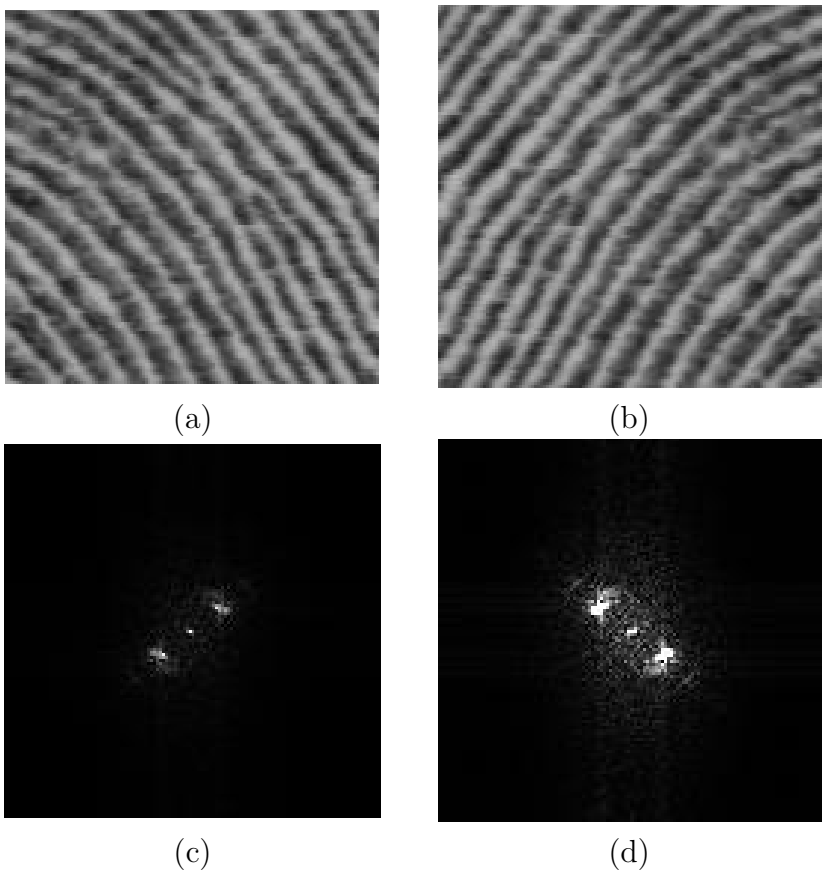
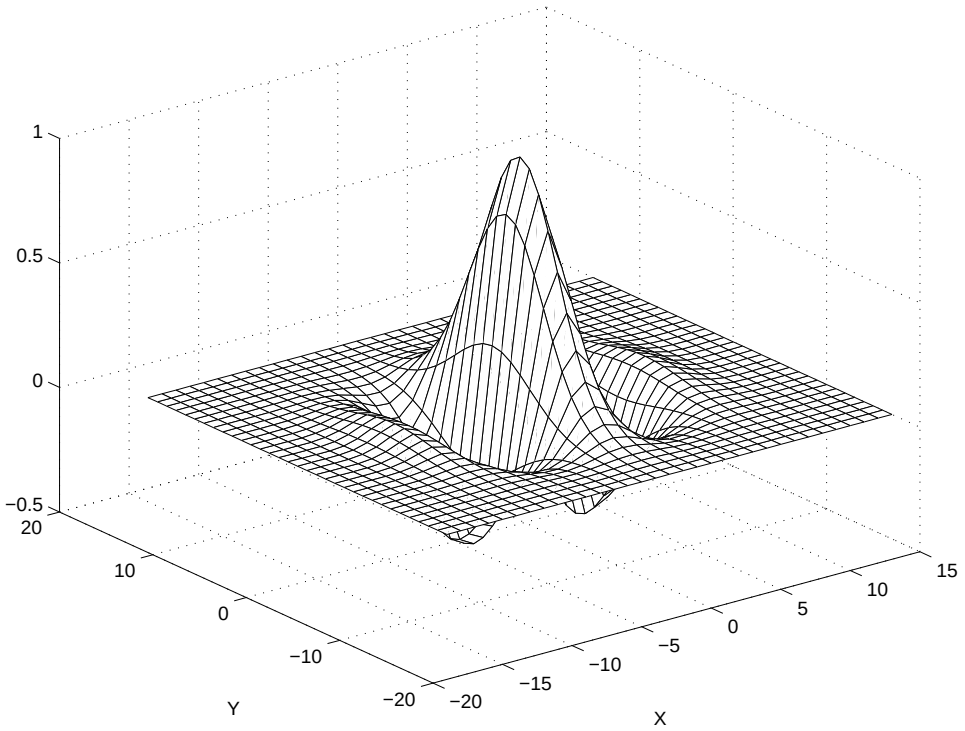


Figure 3.10: Fingerprints have well defined local frequency and orientation. Ridges in local regions are shown in (a) and (b). Fourier spectrum of (a) and (b) are shown in (c) and (d), respectively.

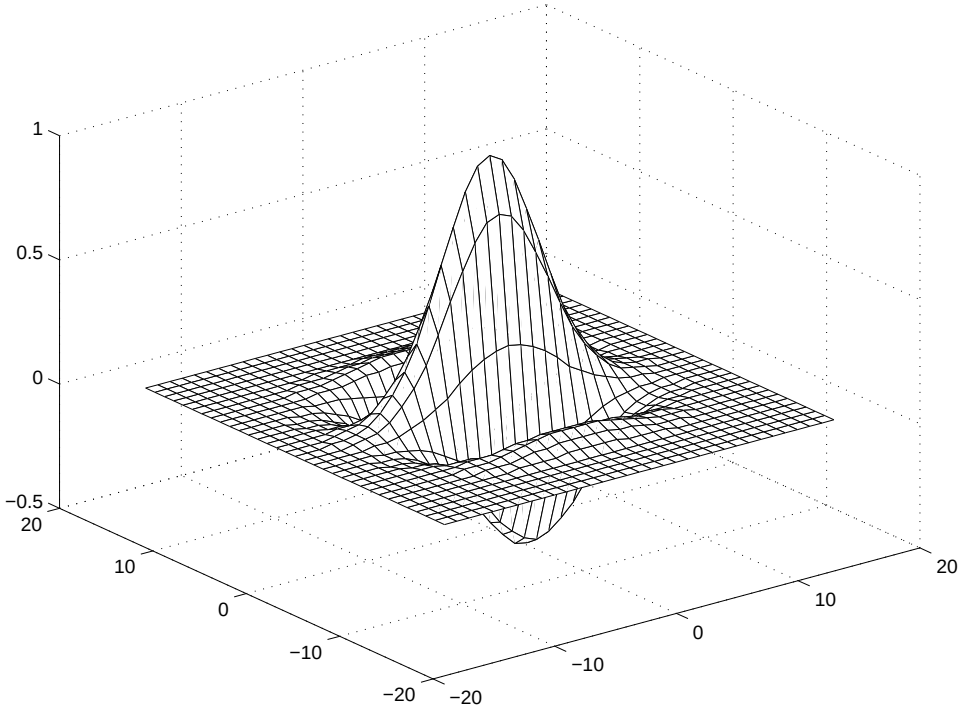
$S_i$, the normalized image is defined as:

$$N_i(x,y) = \begin{cases} M_0 + \sqrt{\frac{V_0 \times (I(x,y)-M_i)^2)}{V_i}}, & if \quad I(x,y) > M_i \\ \\ M_0 - \sqrt{\frac{V_0 \times (I(x,y)-M_i)^2)}{V_i}}, & otherwise, \end{cases} \tag{3.16}$$

where $M_0$ and $V_0$ are the desired mean and variance values, respectively. Normalization is a pixel-wise operation which does not change the clarity of the ridge and valley structures. If normalization is performed on the entire image, then it cannot compensate for the intensity variations in different parts of the image due to the fin-




(a) 0° orientation




(b) 90° orientation

Figure 3.11: Gabor filters (mask size = $33 \times 33$, $f = 0.1$, $\delta_x = 4.0$, $\delta_y = 4.0$). Only 0° and 90° oriented filters are shown here.

ger pressure differences. A separate normalization of each individual sector alleviates this problem. Figure 3.12 shows an example of this normalization scheme. For our experiments, we set the values of both $M_0$ and $V_0$ to 100. The values of $M_0$ and $V_0$ should be the same across all the training and test sets.

An even symmetric Gabor filter has the following general form in the spatial domain:

$$G(x, y; f, \theta) = exp\left\{\frac{-1}{2}\left[\frac{x'^2}{\delta_x^2} + \frac{y'^2}{\delta_y^2}\right]\right\} cos(2\pi f x'), \tag{3.17}$$

$$x' = xsin\theta + ycos\theta, \tag{3.18}$$

$$y' = xcos\theta - ysin\theta, \tag{3.19}$$

where $f$ is the frequency of the sinusoidal plane wave along the direction $\theta$ from the $x$-axis, and $\delta_x$ and $\delta_y$ are the space constants of the Gaussian envelope along $x$ and $y$ axes, respectively. The spatial characteristics of Gabor filters can be seen in Figure 3.11.

We perform the filtering in the spatial domain with a mask size of $33 \times 33$. Figure 3.11 shows that the filter values outside this $33 \times 33$ mask are close to zero. To speed up the filtering process, we convolve a pixel only with those values in the filter mask whose absolute value is greater than 0.05. This speeds up the convolution process significantly while maintaining the information content as the convolution with small values of the filter mask does not contribute significantly to the overall convolution output. We also make use of the symmetry of the filter to speed up the convolution.

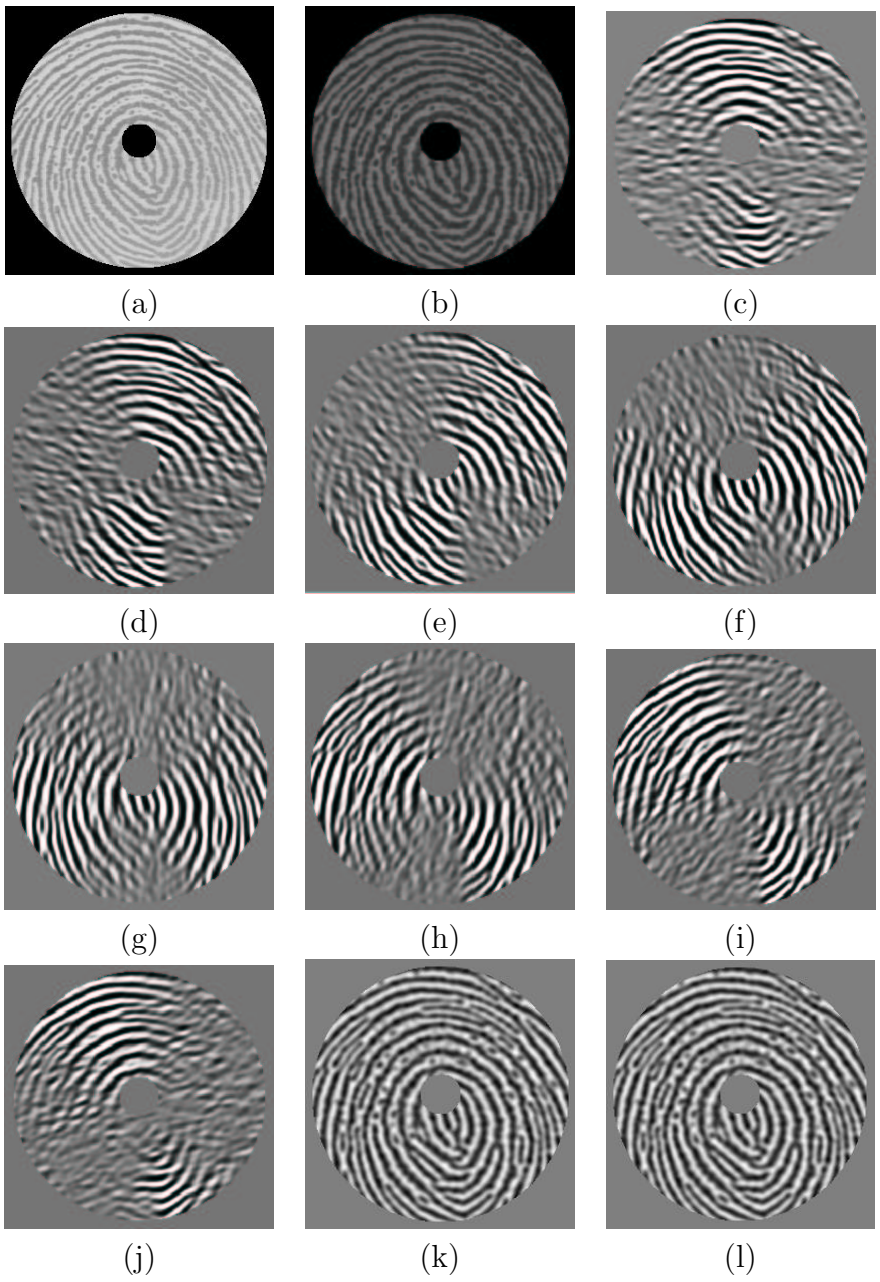



Figure 3.12: Normalized, filtered, and reconstructed fingerprint images. (a) area of interest, (b) normalized image, (c)-(j) 0°, 22.5°, 45°, 90°, 112.5°, 157.5° filtered images, respectively, (k) reconstructed image with 4 filters, and (l) reconstructed image with 8 filters. While four filter orientations are sufficient to capture the global structure of the fingerprint, eight filter orientations are required to capture the local characteristics.

Table 3.1: Gabor filter mask of size $33 \times 33$, $\theta = 0°$, $f = 0.1$, $\delta_x = \delta_y = 4.0$. Only a $19 \times 19$ matrix from the center of the $33 \times 33$ filter is shown because the mask values outside this are zero. Also, only the top left quarter of the mask is shown due to the symmetry in the X and Y axes of the 0° oriented filter. The mask values less than 0.05 are set to zero. Each entry is to be multiplied by $10^{-3}$.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 57 | 62 | 64 | 62 | 57 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | -51 | -59 | -65 | -67 |
| 0 | 0 | -57 | -85 | -120 | -160 | -199 | -232 | -255 | -263 |
| 0 | -62 | -99 | -149 | -210 | -278 | -346 | -404 | -444 | -458 |
| 0 | -66 | -106 | -159 | -224 | -297 | -370 | -433 | -475 | -490 |
| 0 | 0 | -50 | -76 | -106 | -141 | -176 | -205 | -225 | -233 |
| 0 | 0 | 59 | 89 | 125 | 166 | 206 | 241 | 265 | 273 |
| 62 | 106 | 170 | 255 | 359 | 476 | 592 | 692 | 760 | 784 |
| 80 | 135 | 216 | 325 | 458 | 607 | 755 | 882 | 969 | 1000 |

One such mask for the 0°-oriented Gabor filter is shown in Table 3.1. However, convolution with Gabor filters is still the major contributor to the overall feature extraction time (approx. 3 *seconds* of CPU time for convolution of a circular area of radius 120 pixels with 8 Gabor filters on a SUN ULTRA 10 workstation).

In our experiments, we set the filter frequency $f$ to the average ridge frequency $(1/K)$, where $K$ is the average inter-ridge distance. The average inter-ridge distance is approximately 10 pixels in a 500 *dpi* fingerprint image. If $f$ is too large, spurious ridges are created in the filtered image whereas if $f$ is too small, nearby ridges are merged into one. Different filter directions ($\theta$) include 0°, 22.5°, 45°, 67.5°, 90°, 112.5°, 135°, and 157.5° with respect to the $x$-axis. The normalized region of interest in a fingerprint image is convolved with each of these eight filters to produce a set of eight filtered images. A fingerprint convolved with a 0°-oriented filter accentuates those ridges which are parallel to the $x$-axis and smoothes the ridges in the other

directions. Filters tuned to other directions work in a similar way. These eight directional-sensitive filters capture most of the global ridge directionality information as well as the local ridge characteristics present in a fingerprint. We illustrate this through reconstructing a fingerprint image by adding together all the eight filtered images. The reconstructed image is similar to the original image without a significant loss of information (Figure 3.12(l)). Empirically, we have determined that at least four directional filters are required to capture the entire global ridge information in a fingerprint (Figure 3.12(k)), but eight directional filters are required to capture the local characteristics. By capturing both the global and local information, the verification accuracy is improved although there is some redundancy among the eight filtered images. If $\delta_x$ and $\delta_y$ (standard deviations of the Gaussian envelope) values are too large, the filter is more robust to noise, but is more likely to smooth the image to the extent that the ridge and valley details in the fingerprint are lost. If $\delta_x$ and $\delta_y$ values are too small, the filter is not effective in removing the noise. The values for $\delta_x$ and $\delta_y$ were empirically determined and each is set to 4.0 (about half the average inter-ridge distance).

## 3.5 Feature Vector

It is difficult to rely on features that are extracted based on explicit detection of structural features in fingerprints, especially in poor quality images. Features based on statistical properties of images are likely to degrade gracefully with the image quality deterioration. For this study, we use grayscale variance-based features. The

average absolute deviation of the gray levels from the mean value in an image sector is indicative of the overall ridge activity in that sector which we claim to be useful for fingerprint classification and verification. Similar features were successfully used earlier by Jain and Farrokhnia [10] for texture classification and segmentation. Our empirical results on fingerprint classification and verification applications show that this simple statistical feature performs extremely well.

Let $F_{i\theta}(x, y)$ be the $\theta$-direction filtered image for sector $S_i$. Now, $\forall\ i \in \{0, 1, \ldots, 79\}$ and $\theta \in \{0°, 22.5°, 45°, 67.5°, 90°, 112.5°, 135°, 157.5°\}$, the feature value, $V_{i\theta}$, is the average absolute deviation from the mean defined as:

$$V_{i\theta} = \frac{1}{n_i} \left( \sum_{n_i} |F_{i\theta}(x, y) - P_{i\theta}| \right), \tag{3.20}$$

where $n_i$ is the number of pixels in $S_i$ and $P_{i\theta}$ is the mean of pixel values of $F_{i\theta}(x, y)$ in sector $S_i$. The average absolute deviation of each sector in each of the eight filtered images defines the components of our 640-dimensional feature vector. The feature vectors for some example images in the MSU_DBI database are shown as grayscale images in Figure 3.13.

The average absolute deviation (AAD) features give slightly better performance than variance features in our experiments. The number of filter orientations required was empirically determined. In the fingerprint verification application, using eight orientation filters resulted in a better performance than when only four orientation filters were used. A further increase in the number of filters did not provide any increase in the verification performance. Similarly, using eight filters instead of four

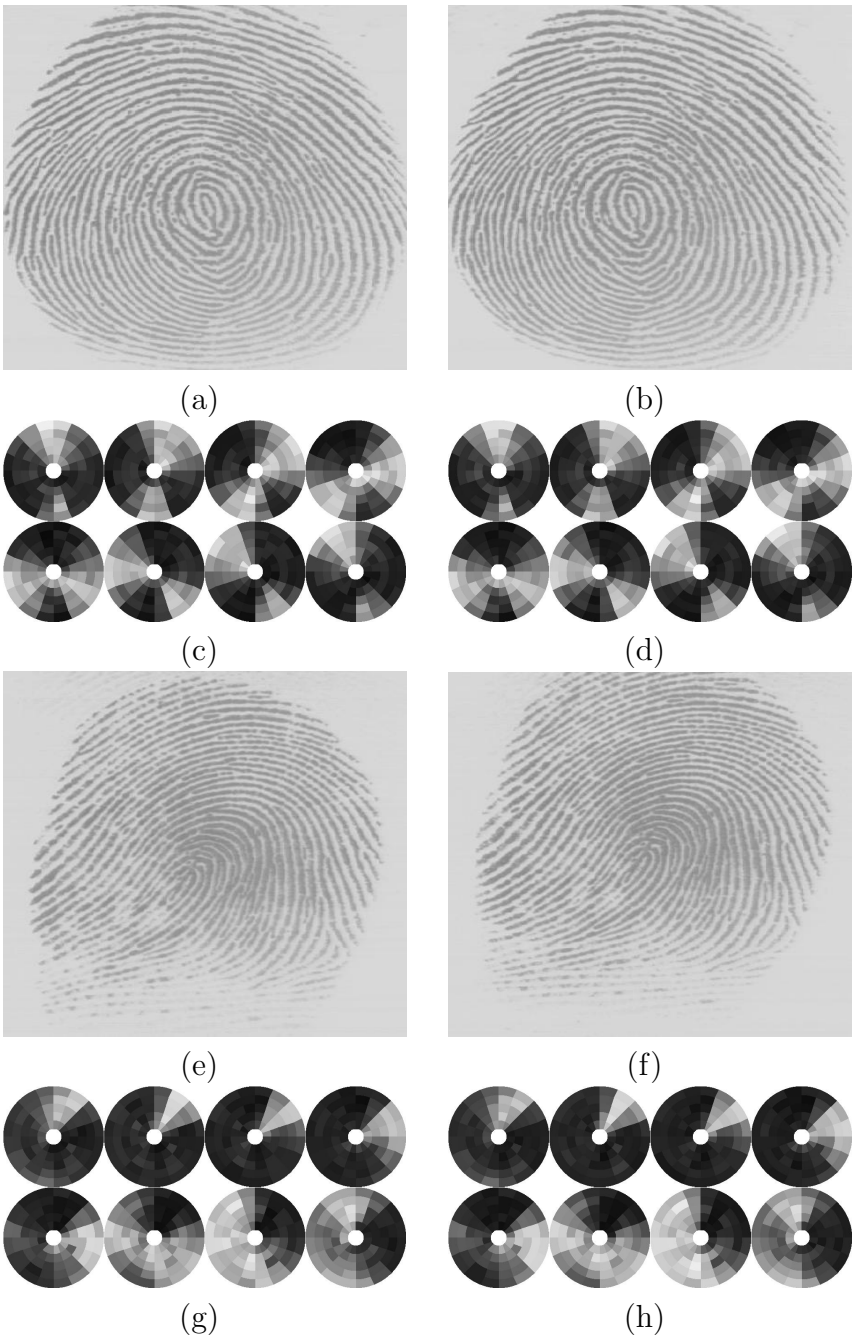



Figure 3.13: Examples of 640-dimensional feature vectors. (a) First impression of finger 1, (b) Second impression of finger 1, (c) and (d) are the corresponding FingerCodes, (e) First impression of finger 2, (f) Second impression of finger 2, (g) and (h) are the corresponding FingerCodes.

filters did not improve the performance of the fingerprint classification algorithm (see Chapter 4.

The 640-dimensional feature vectors (FingerCodes) for fingerprint images of two different fingers from the MSU_DBI database are shown as gray level images with eight disks, each disk corresponding to one filtered image in Figure 3.13. The gray level in a sector in a disk represents the feature value for that sector in the corresponding filtered image. Note that Figures 3.13(c) and (d) appear to be visually similar as are Figures 3.13(g) and (h), but the corresponding disks for two different fingers look very different.

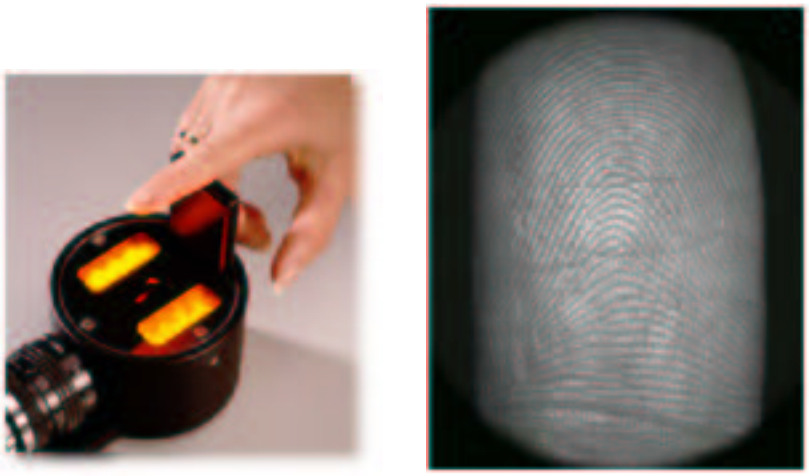
The translation is handled by a *single* reference point location during the feature extraction stage. Our representation scheme is able to tolerate the imprecision in the reference point estimates of up to 10 pixels (approximately 1 inter-ridge distance unit) away from its "true" location. A circular tessellation is chosen because the sector size increases as we go farther away from the center and handles the error in center location better. The present implementation of feature extraction assumes that the fingerprints are vertically oriented (fingertip pointed straight up). In reality, the fingerprints in our database are not exactly vertically oriented; the fingerprints may be oriented up to $\pm 45^{\circ}$ away from the assumed vertical orientation. The circular tessellation assists in obtaining a representation corresponding to a rotation of the fingerprint image by a cyclic rotation of the values in the feature vector. We use this cyclic rotation of the feature vector to partially handle the rotation in the matching stage.

## 3.6 Summary

Chapter 2 establishes an upper bound on the performance of the minutiae-based automatic fingerprint identification systems due to the limited information content of the minutiae representation. This was a powerful motivation for exploring a novel and rich alternate representation for fingerprints. The proposed filterbank-based representation for fingerprints was motivated by Daugman's work on Iris recognition [86] that quantified the textural information present in the human iris using a Gabor filterbank in a small IrisCode. Gabor filterbank has also been successfully used in texture classification and segmentation tasks [10]. Since fingerprint images can be viewed as a textured pattern, it is appropriate to use this filterbank-based representation for fingerprints. Our proposed filterbank-based representation has the desirable property of capturing both the local minute details and the global pattern information in a fingerprint. One of the main advantages of this representation is that a single representation can be used for fingerprint classification as well as matching. As a comparison, earlier approaches to fingerprint representation are either exclusively local (e.g., minutiae) or exclusively global (e.g., orientation field). The exclusively local representation is traditionally used for fingerprint matching while the exclusively global representation is used for fingerprint classification. Additionally, the compactness of the filterbank representation is very attractive for credit card or smart card applications where the amount of available storage is limited. The good discriminatory power of the representation is demonstrated by the classification and verification applications in Chapters 4 and 5, respectively. The current implementation of the

feature extraction is computationally expensive due to the image convolution operations. It is possible to significantly enhance the speed of the feature extraction algorithm by implementing the convolution operation via a dedicated DSP chip. For an example, a DSP implementation of the FingerCode extraction algorithm using an Analog Devices Sharc (Super Harvard Architecture Computer) DSP 21062 EZ-LAB Development board developed by Bittware, Inc. was reported in [115] and the feature extraction time was reduced by an order of magnitude. The primary advantage of our approach is its computationally attractive matching/indexing capability. As far as the fingerprint database is concerned, the feature extraction is an off-line process and if the normalized (for orientation and size) FingerCodes of all the enrolled fingerprints are stored as templates, the classification or verification effectively involves a "bit" comparison with the test image. As a result, the identification time would be relatively insensitive to the database size because "bit" comparison is an extremely fast operation. Further, our approach for representation extraction and matching is more amenable to hardware implementation than, say, a string-based fingerprint matcher.

There are a number of limitations of our approach to fingerprint representation. The implementation of the representation extraction algorithm that is based on a reference point in the fingerprint image rejects about 5% of the images (in the NIST-9 database) due to failure of the reference point location on poor quality fingerprint images. An alignment based on the minutiae points or orientation field in an image is expected to overcome this problem but the resulting representation is not translation and rotation invariant and thus, is not very attractive for indexing purposes. More-



(a) (b)

Figure 3.14: Example of new touchless fingerprint sensor TFS 050 from Biometric Partners, Inc. (http://www.biometricpartners.com/). The touchless sensor captures a fingerprint from a distance of approximately 50mm. Advantages of touchless technology include capture of larger fingerprint area, is more hygienic, the sensor does not degrade with repeated use, and there is no nonlinear distortion due to finger pressure difference in the captured image. The image captured by the sensor in (a) is shown in (b). However, the touchless sensors have their own problems, including poor quality images.

over, the filterbank representation is not invariant to nonlinear deformations which is an inherent property of the touch-based fingerprint sensing process. The new generation of touchless sensors (see Figure 3.14) do not suffer from nonlinear deformations in the captured fingerprint image but there are additional degrees of freedom in the translation (translation in $z$-axis results in a scaling in two-dimensional projection) and rotation variance.

# Chapter 4

# Fingerprint Classification

Fingerprint classification provides an important indexing mechanism in a fingerprint database. An accurate and consistent classification can greatly reduce fingerprint matching time for a large database. We present a fingerprint classification algorithm which is able to achieve an accuracy which is comparable to the algorithms reported in the literature. In 1899, Edward Henry and his two assistants established the "Henry System" of fingerprint classification [78]. The Henry system classifies fingerprints into three main categories: (*i*) loop, (*ii*) whorl, and (*iii*) arch. Each category is then further divided resulting in a total of more than twenty categories. Federal Bureau of Investigation (FBI) follows Henry system of classification but recognizes only eight different types of fingerprint: radial loop, ulnar loop, double loop, central pocket loop, plain arch, tented arch, plain whorl, and accidental. Due to the small interclass separability of these types fingerprint types, it is extremely difficult to design an eight-class classifier with high accuracy. As a result, most automatic systems reduce the number of fingerprint types to a subset of classes defined in the Henry system. For

example, academic institutes have typically concentrated on a five-class classification that includes whorl, left loop, right loop, arch, and tented arch, while the commercial systems typically provide ulnar, radial loops, accidental, whorl, double loop, and arch classification [78].

Fingerprint classification remains a very difficult problem for both human experts and automatic systems because of large variations in fingerprint configurations. A substantial amount of experience is required for a forensic expert to reach a satisfactory level of performance in fingerprint classification. Fingerprints have a continuum in the pattern space. For example, there is a continuum of patterns between the two extremes of a "true" arch and a "true" loop. As a result, there exists patterns which lie on any arbitrarily drawn class boundary drawn for an exclusive classification. Due to the fuzzy boundaries between the large number of fingerprint classes, NIST [41] chose a five-element subset in the Henry system of fingerprint classification for automatic system development. These five classes are whorl, right loop, left loop, arch, and tented arch. Our automatic system classifies fingerprints into these five categories. The algorithm uses the novel representation (FingerCode) described in Chapter 3 and is based on a two-stage classifier to make a decision. Our approach has been tested on $4,000$ images in the NIST-4 database. For the five-class problem, a classification accuracy of 90% is achieved (with a 1.8% rejection during the feature extraction phase). For the four-class problem (arch and tented arch combined into one class), we are able to achieve a classification accuracy of 94.8% (with 1.8% rejection). By incorporating a reject option in the classifier, the classification accuracy can be increased to 96% for the five-class classification task, and to 97.8% for the

four-class classification task after a total of 32.5% of the images are rejected.

## 4.1 Introduction

Several approaches have been developed for automatic fingerprint classification. These approaches can be broadly categorized into four main categories: (*i*) knowledge-based, (*ii*) structure-based, (*iii*) frequency-based, and (*iv*) syntactic. The knowledge-based fingerprint classification technique uses the locations of singular points (core and delta) to classify a fingerprint into the five above-mentioned classes [97, 105]. A knowledge-based approach tries to capture the knowledge of a human expert by deriving rules for each category by hand-constructing the models and therefore, does not require training. Accuracies of 85% [97] and 87.5% [105] have been reported on the NIST-4 database [41] using these approaches. A structure-based approach uses the estimated orientation field in a fingerprint image to classify the fingerprint into one of the five classes. An accuracy of 90.2% with 10% rejection is reported on NIST-4 [43]. The neural network used in [43] was trained on images from $2,000$ fingers (one image per finger) and then tested on an independent set of $2,000$ images taken from the same fingers. The error reported is thus optimistically biased. A *later* version of this algorithm [76] was tested on the NIST-14 database which is a *naturally* distributed database resulting in a better performance (in a naturally distributed database, the number of fingerprint images for a particular fingerprint type is proportional to the probability of occurrence of that type in nature). A further enhancement of this algorithm was reported in [44, 45]. However, this performance

improvement should be expected since the NIST-14 database contains only a small percentage of arch-type fingerprints which pose the most difficultly for fingerprint classifiers, and the neural network used in the algorithm implicitly takes advantage of this information. A similar structure-based approach which uses hidden Markov models for classification [30] depends on a reliable estimation of ridge locations which is difficult in noisy images. In another structure-based approach, B-spline curves are used to represent and classify fingerprints [122]. A syntactic approach uses a formal grammar to represent and classify fingerprints [46]. Frequency-based approaches use the frequency spectrum of the fingerprints for classification [25]. Hybrid approaches combine two or more approaches for classification [34, 120]. These approaches show some promise but have not been tested on large databases. For example, Chong et al. [122] report results on 89 fingerprints, Fitz and Green [25] on 40 fingerprints, and Kawagoe and Tojo [120] on 94 fingerprints. Recently, Cappelli et al. [141] proposed a fingerprint classification algorithm based on the multi-space KL transform applied to the orientation field. This algorithm reports about 2% better accuracy than our algorithm on the NIST-4 database. See Table 4.1 for a comparison of different fingerprint classification algorithms.

Most of the information about a fingerprint category is contained in the central part of the fingerprint, called the *pattern area* [68]. The pattern area is the area between the two innermost ridges (known as *typelines*) that form a divergence tending to encircle or encompass the central portion of the fingerprint as shown in Figure 4.1. The knowledge-based techniques which use both the core and delta points for classification require that these singular points be present in the image. The dab fingerprint

Table 4.1: Fingerprint classification literature survey. The number of classes is denoted by $C$, the classification accuracy is denoted by $Acc$, and the reject rate is denoted by $RR$. The classification accuracies reported by the different authors are on different databases with different number of fingerprints and therefore, they cannot be directly compared. Most of the work in fingerprint classification is based on supervised learning and discrete class assignment using knowledge-based features.

| Authors | $C$ | Features | Method | $Acc.$ ($RR$) |
|---|---|---|---|---|
| Kawagoe and Tojo 1984 | 7 | Singular points | Rule-based | 91.5% (0%) |
| Blue et al. 1994 | 5 | Orientation field | Neural network | 92.8% (0%) |
| Wilson et al. 1994 | 5 | Orientation field | Neural network | 90.2% (10%) |
| Candela et al. 1995 | 6 | Orientation field | Neural network | 92.2% (0%) |
| Pal and Mitra 1996 | 5 | Orientation field | Neural network | 82+% (0%) |
| Fitz and Green 1996 | 3 | FFT | Nearest-neighbor | 85 % (0%) |
| Karu and Jain 1996 | 5 | Singular points | Rule-based | 85% (0%) |
| Senior 1997 | 4 | Ridge lines | Hidden Markov Model | 90% (0%) |
| Chong et al. 1997 | 5 | Ridge lines | Rule-based | 96.5% (0%) |
| Hong and Jain 1999 | 5 | Singular points and ridge lines | Rule-based | 87.5% (0%) |
| Proposed 1999 | 5 | Gabor response | Combination | 90% (1.8%) |
| Cappelli et al. 2000 | 5 | Orientation field | Combination | 99% (20%) |

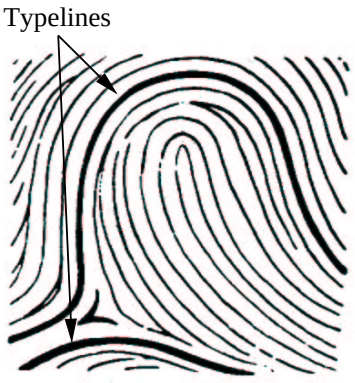



Figure 4.1: Pattern area and typelines [68, 104].

images obtained by optical scanners do not always capture the entire fingerprint and often have the delta point(s) missing. Also, the core or delta point(s) are difficult to detect in noisy fingerprint images. There is, however, sufficient information available in the ridge pattern itself to classify a fingerprint. While the structure-based approach does not depend upon the core or delta points, it requires a reliable estimate of the orientation field which is very difficult to obtain in low quality fingerprint images.

We propose a fingerprint classification algorithm (Figure 4.2) based on our filterbank fingerprint representation scheme which is directly derived from local ridge structures. The representation does not use the core, delta, and orientation field, explicitly. It is more capable of tolerating poor image quality, which is a major difficulty in fingerprint classification. The main steps of our classification algorithm are as follows: (*i*) Locate a reference point in the input image and define a spatial tessellation (sectors) of the region around the reference point; (*ii*) decompose the input image into a set of component images, each of which preserves certain ridge orientation information; compute the standard deviation of the component images in each sector to generate the feature vector (called FingerCode); (*iii*) feed the feature

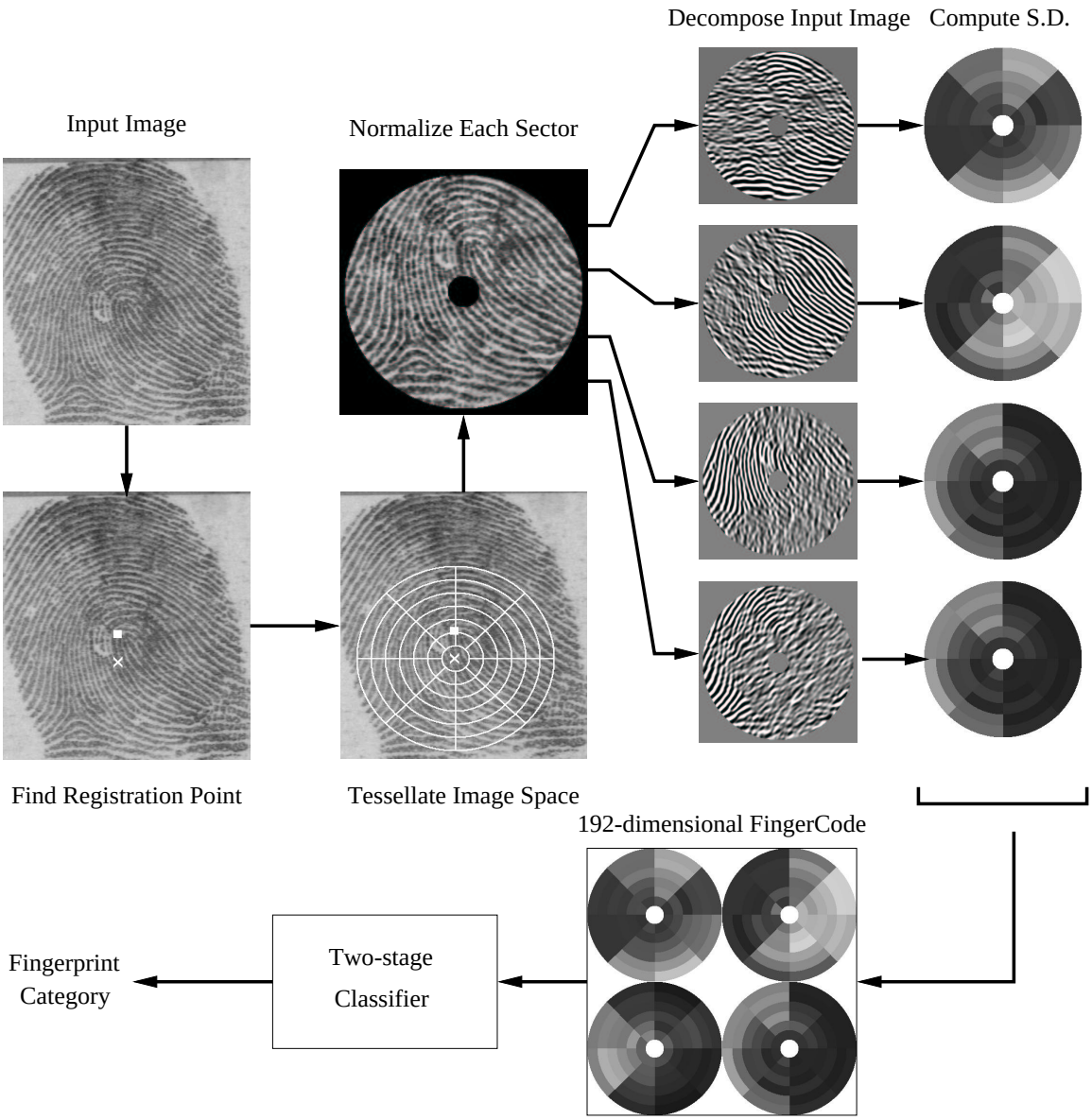



Figure 4.2: Flow diagram of our fingerprint classification algorithm.

vector into a multi-stage classifier; in our algorithm, a two-stage classifier is used. This two-stage classifier uses a $K$-nearest neighbor classifier in its first stage and a set of neural network classifiers in its second stage to classify a feature vector into one of the five fingerprint classes.

In the following sections, we will present the details of our fingerprint classification algorithm.

## 4.2 Feature Extraction

The category of a fingerprint is determined by its global ridge and furrow structures. A valid feature set for fingerprint classification should be able to capture this global information effectively. The filterbank-based fingerprint representation developed in Chapter 3 is able to represent both the minute details and the global ridge and furrow structures of a fingerprint. For the purpose of classification, we adapt our representation such that it is very effective in representing the global ridge and furrow structures and is invariant to individual minute details.

The representation scheme developed in Chapter 3 has certain parameters which are adapted to our fingerprint classification algorithm. We choose the tessellation parameter $B$, the number of concentric bands to be 6 based on the size of the images in the NIST 4 database ($512 \times 512$). Number of sectors in each band is chosen to be eight ($k = 8$). This results in large sectors which are capable of capturing the global information in the fingerprints. Thus, a total of $8 \times 6 = 48$ sectors ($S_0$ through $S_{47}$) are defined. Since most of the category information is present in the part below the

core point in the fingerprints (different fingerprint types have similar ridge structure in the part above the core point), we move the reference point down by 40 pixels with respect to the core point detected by the reference point detection algorithm in Chapter 3.

Figure 4.3: Reference point detected by the algorithm described in Chapter 3 (□), moved reference point (×), the region of interest and 48 sectors.

A fingerprint image is convolved with four Gabor filters ($\theta = 0°$, $45°$, $90°$, and $135°$) to produce the four component images. Thus, our feature vector is 192-dimensional ($48 \times 4$). Our experimental results indicate that the four component images capture most of the ridge directionality information present in a fingerprint image and thus form a valid representation. We illustrate this by reconstructing a fingerprint image by adding together all the four filtered images. The reconstructed image is similar

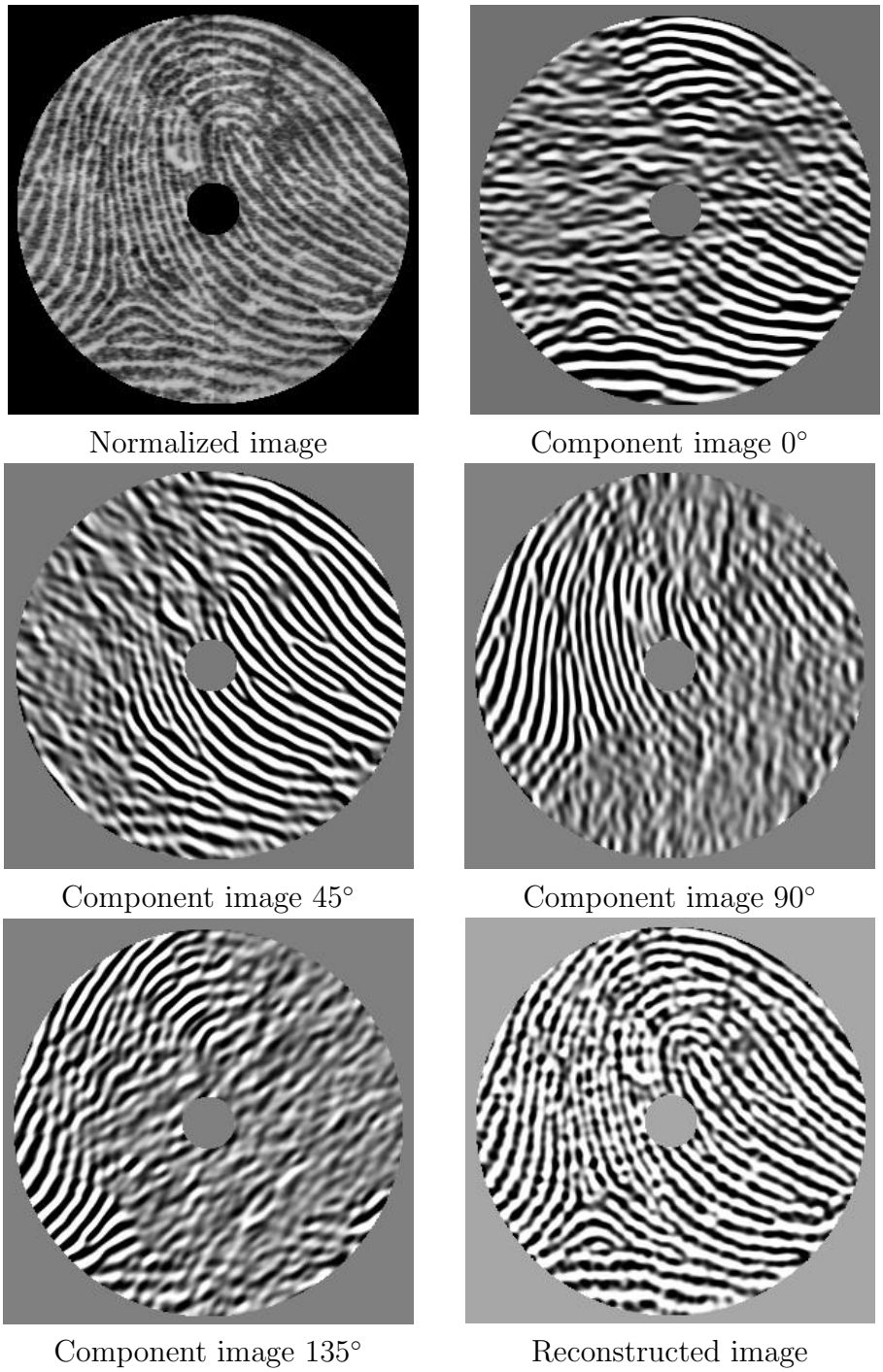


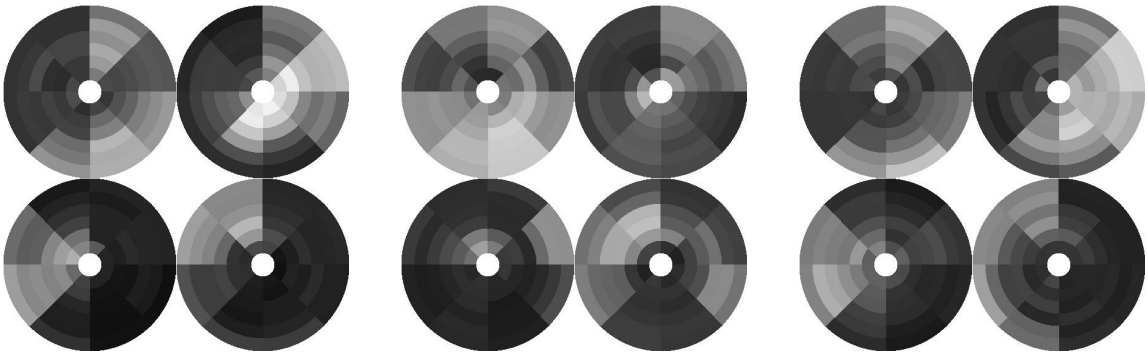

Figure 4.4: Normalized, filtered, and reconstructed fingerprint images.

(a) (b)

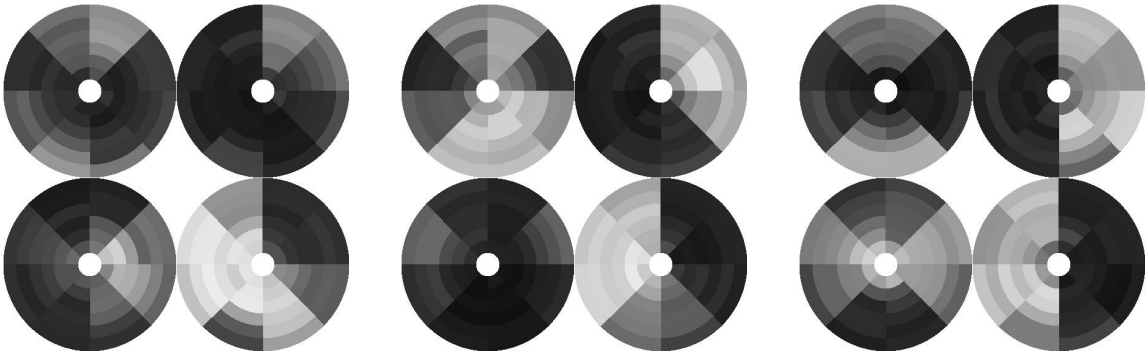
Figure 4.5: Reconstructed fingerprint images using (a) four filters, and (b) eight filters. Most of the directionality information is captured by four filters.

to the original image without a significant loss of information (Figure 4.4). Using additional filters does not necessarily improve the directionality information in the reconstructed image (see the comparison of reconstruction using four filters with reconstruction using eight filers in Figure 4.5). Since convolution with Gabor filters is an expensive operation, the use of additional filters will increase the classification time without necessarily improving the classification accuracy.

In each component filtered image, a local neighborhood with ridges and furrows that are parallel to the corresponding filter direction exhibits a higher variation, whereas a local neighborhood with ridges and furrows that are not parallel to the corresponding filter tends to be diminished resulting in a lower variation. The spatial distribution of the variations in local neighborhoods of the component images thus constitutes a characterization of the global ridge structures which is captured by the average absolute deviation of grayscale values from the mean ($AAD$ features) (Equation (3.20)).



(a) Test Image (b) Whorl (c) Right Loop



(d) Left Loop (e) Arch (f) Tented Arch

Figure 4.6: Fingerprint representation using 192-dimensional feature vectors (In each representation, the top left disc represents the 0° component, the top right disc represents the 45° component, the bottom left disc represents the 90° component, and the bottom right disc represents the 135° component). The test image is a right loop. Each disk corresponds to one particular filter and there are 48 features (shown as gray values) in each disk ($8 \times 6 = 48$ sectors) for a total of 192 ($48 \times 4$) features.

## 4.3 Classification

Automatic classification of fingerprints is a difficult problem because of the small *interclass* variability and large *intraclass* variability among the five classes under consideration. In order to simplify the classification task, we decompose the five-class problem into a set of 10 two-class problems. Further, we use a two-stage classifier for fingerprint classification. In the first stage, we use a $K$-nearest neighbor classifier to find the two most probable classes for a given input pattern. The $K$-nearest neighbor

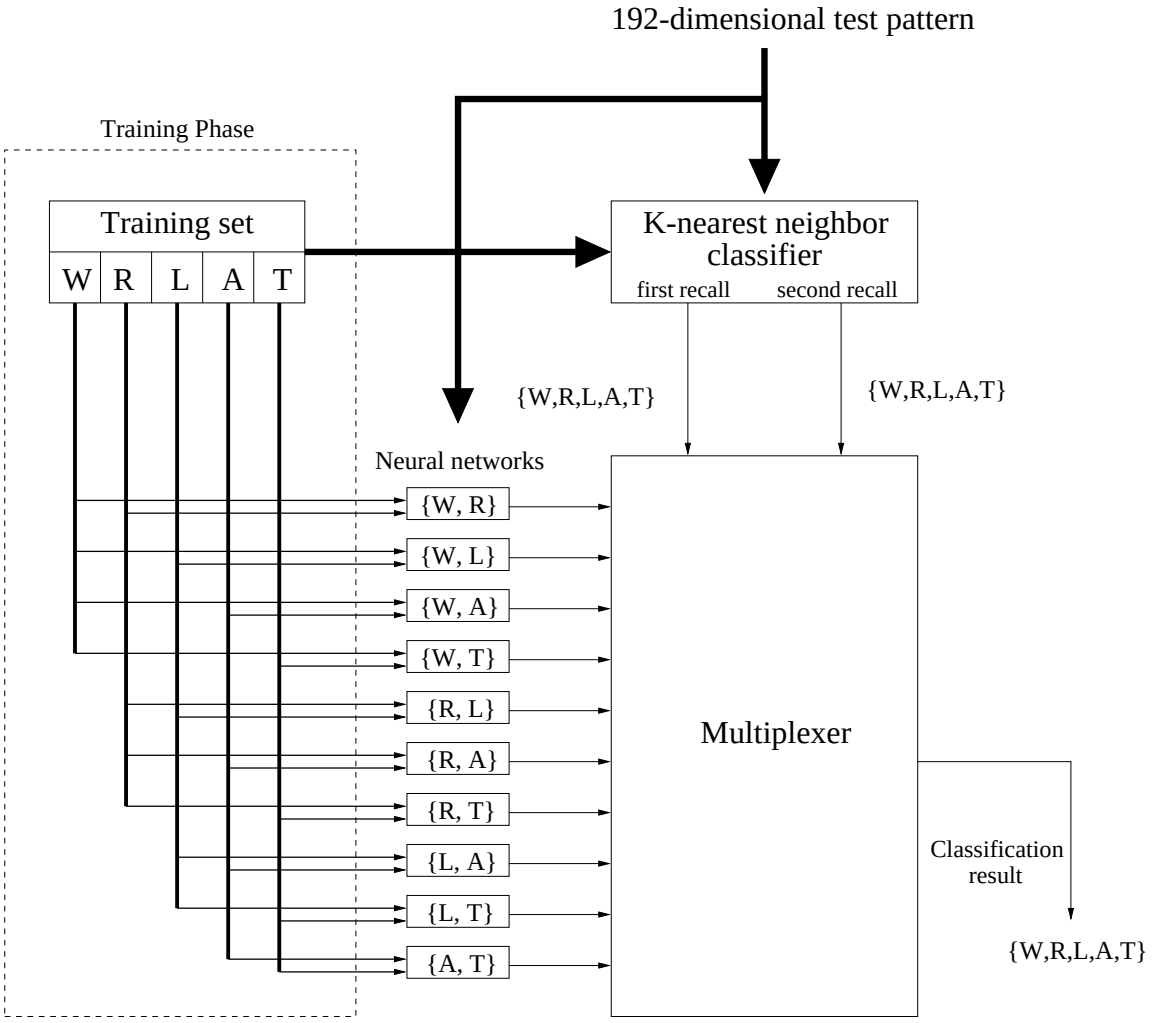


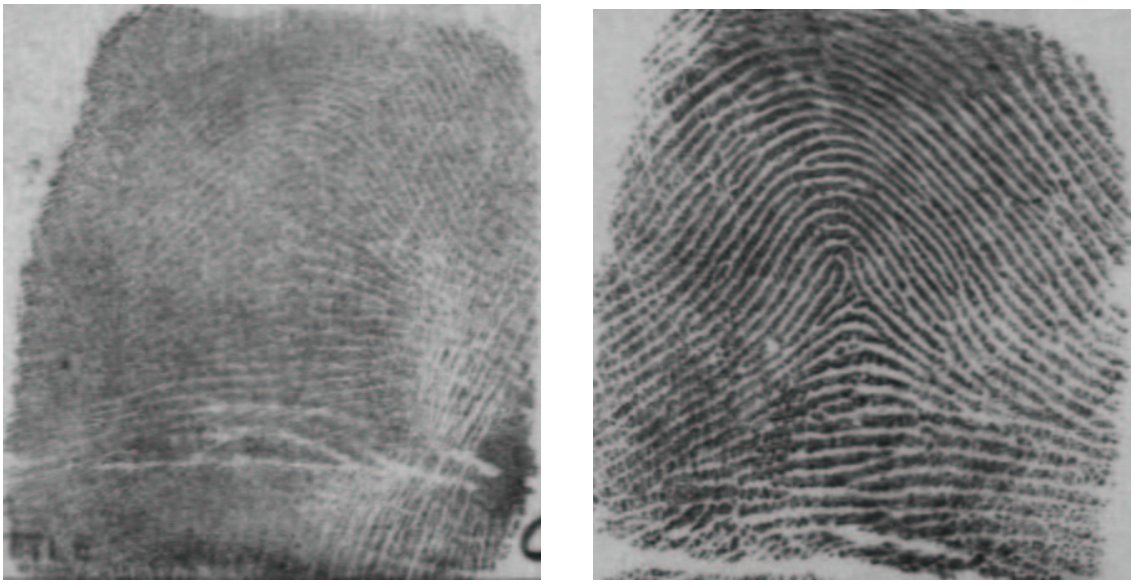

Figure 4.7: Two-stage classification scheme using K-NN and neural network classifiers.

decision rule first finds the $K$ nearest neighbors of the test pattern in the feature space and then assigns the test pattern to the class which is most frequently represented among the $K$ nearest neighbors. The top two categories can be retrieved from the $K$-NN classifier corresponding to the classes which have the highest and the second highest count among the $K$ nearest neighbors, i.e., the first recall and the second recall. In the second stage of the classifier, 10 ($C_2^5$) neural networks are trained to solve each of the 10 two-class problems. The second stage uses the first and second recalls to select the specific neural network which has been trained to distinguish

between the corresponding pair of classes and the input pattern is then sent to the selected neural network for further classification. This neural network makes the final decision between these two classes.

## 4.4 Experimental Results

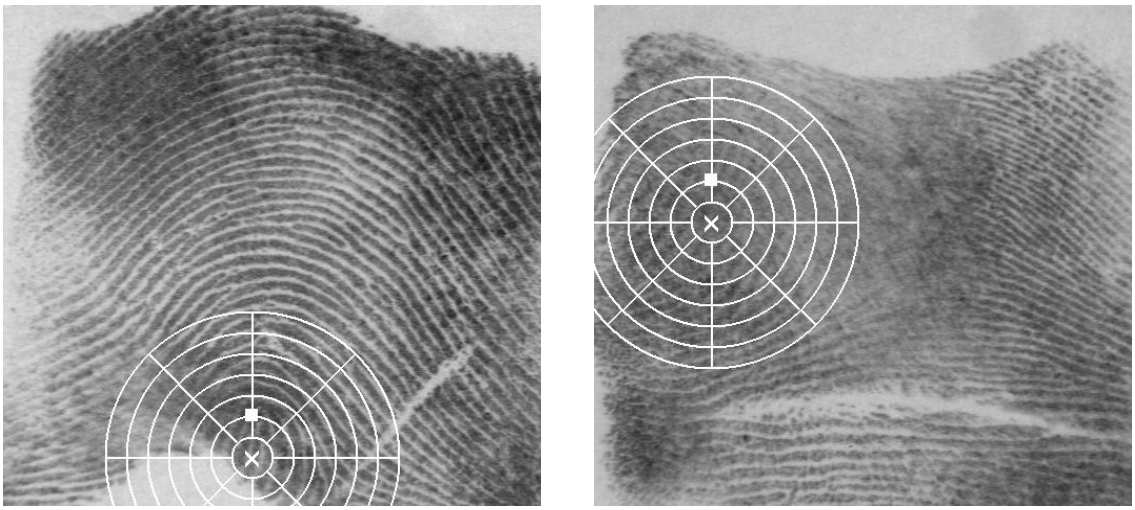
### 4.4.1 Dataset



(a) (b)

Figure 4.8: Example of images in the NIST 4 database with two ground truth labels. The poor quality fingerprint in (a) is labeled as belonging to both the arch and tented arch classes, (b) is labeled as belonging to both the left loop and tented arch classes.

The NIST-4 database consists of $4,000$ fingerprint images (image size is $512 \times 480$) from $2,000$ fingers. Each finger has two impressions ($first$ and *second*). Each image is labeled with *one or more* of the five classes ($W$, $R$, $L$, $A$, and $T$). About 17% of the fingerprint images in the NIST 4 database are labeled with two labels which shows that there is disagreement among the human experts about the true class of a large



(a) (b)

Figure 4.9: Example of images which were rejected because a valid tessellation could not be established.

number of fingerprints. The fraction of images with more than one label can also be interpreted as a measure of human accuracy in classifying fingerprints. On this basis, we can say that there is about 17% error in classifying fingerprints by human experts. The accuracy of the current automatic fingerprint classification system is of the same order. See Figure 4.8 for examples of fingerprint images that were assigned two different labels. To simplify the training procedure, we make use of only the first label of a fingerprint to train our system. For testing, however, we make use of all the true labels assigned to a fingerprint and consider the output of our classifier to be correct if the output matches any one of the labels. This is in line with the common practice used by other researchers in comparing the classification results on the NIST-4 database. The images in the NIST-4 database are numbered $f0001$ through $f2000$ and $s0001$ through $s2000$. Each number represents a fingerprint from a different finger. We form our training set with the first $2,000$ fingerprints from $1,000$ fingers ($f0001$ to $f1000$ and $s0001$ to $s1000$) and the test set contains the remaining $2,000$

fingerprints ($f1001$ to $f2000$ and $s1001$ to $s2000$). The natural proportion (prior probabilities) of fingerprints belonging to each class is 0.279, 0.317, 0.338, 0.037, and 0.029 for the classes W, R, L, A, and T, respectively [43]. Classification accuracies can be significantly increased by using datasets whose records follow the natural distribution of fingerprint classes because the more common types of fingerprints (*loop* and *whorl*) are easier to recognize. However, we do not use datasets with a natural class distribution. Twenty eight fingerprints from the training set were rejected by our feature extraction algorithm because the reference point was detected at a corner of the image and, therefore, a valid tessellation could not be established for these images (Figure 4.9). Thirty five fingerprints were rejected from the test set for the same reason. So, our training set contains $1,972$ fingerprint images and the test set contains $1,965$ fingerprint images. The thirty five images rejected from the test set of $2,000$ fingerprints amounts to a reject rate of 1.8%. We report the results of our fingerprint classification algorithm on the NIST-4 database for the five-class fingerprint classification problem. Since fingerprint classes $A$ (*arch*) and $T$ (*tented arch*) have a substantial overlap, it is very difficult to separate these two classes. Therefore, we also report our results for the four-class classification problem, where classes $A$ and $T$ have been merged into one class. By incorporating a rejection option, classification accuracy can be increased. We report the improvement in error rates at different rejection rates for both the five-class and the four-class classification problems.

### 4.4.2 $K$-Nearest neighbor classifier

The $K$-nearest neighbor classifier results in an accuracy of 85.4% for the five-class classification task when 10 nearest neighbors ($K = 10$) are considered. Classification accuracy does not always increase with increasing $K$; there exists an optimal value of $K$ which is a function of the number of available training samples (Figure 4.10) [8]. For the four-class classification task (where classes $A$ and $T$ were collapsed into one class), an accuracy of 91.5% is achieved. The confusion matrix for the $K$-nearest neighbor classification for the five-class problem is shown in Table 4.2. The diagonal entries in this matrix show the number of test patterns from different classes which are correctly classified and the off-diagonal entries denote the number of classification errors. Since a number of fingerprints in the NIST-4 database are labeled as belonging to two different classes, row sums of the confusion matrices in Tables 4.2, 4.4, and 4.6 are not identical.

Table 4.2: Confusion matrix for the $K$-nearest neighbor classification for the five-class problem; $K = 10$.

| True class | Assigned class | | | | |
|---|---|---|---|---|---|
| | W | R | L | A | T |
| W | **320** | 38 | 31 | 6 | 0 |
| R | 1 | **368** | 2 | 10 | 21 |
| L | 0 | 1 | **359** | 13 | 8 |
| A | 1 | 3 | 7 | **422** | 20 |
| T | 0 | 15 | 16 | 95 | **208** |

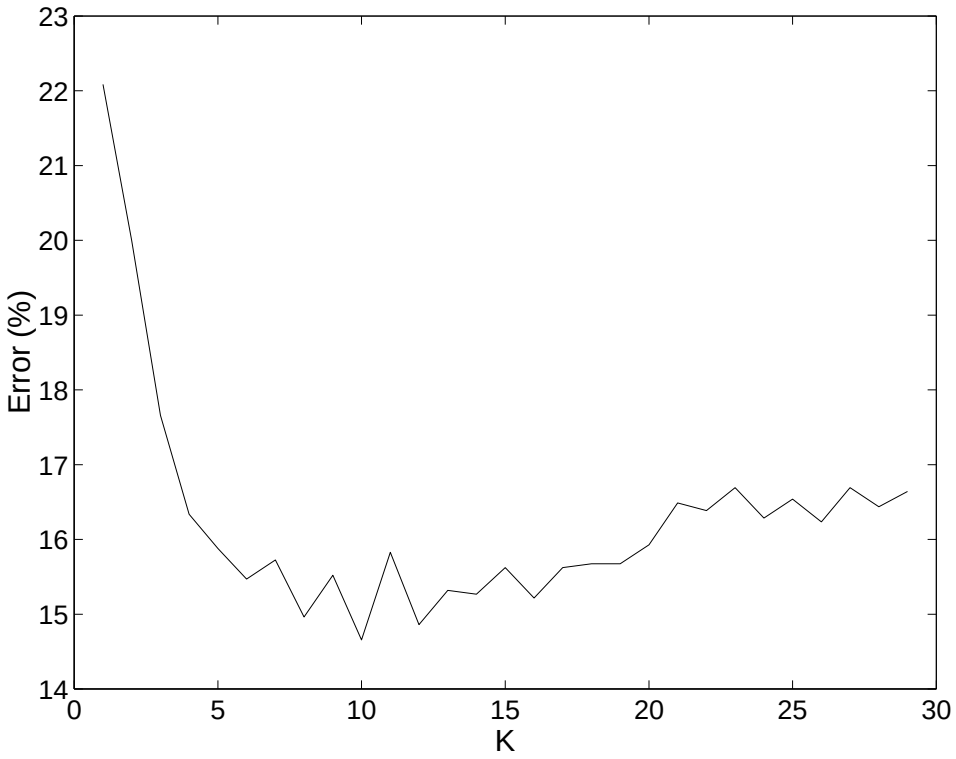



Figure 4.10: $K$ vs. classification error for the $K$-nearest neighbor classifier for the five-class problem.

### 4.4.3 Neural network classifier

We trained a multi-layer feed-forward neural network using a quick propagation training algorithm [154]. The neural network has one hidden layer with 20 neurons, 192 input neurons corresponding to the 192 features, and 5 output neurons corresponding to the five classes. We obtained an accuracy of 86.4% for the five-class classification task. For the four-class classification task, an accuracy of 92.1% is achieved. The confusion matrix for the neural network classification is shown in Table 4.4.

Table 4.3: Confusion matrix for the $K$-nearest neighbor classification for the four-class problem; $K = 10$.

| True class | Assigned class | | | |
|---|---|---|---|---|
| | W | R | L | A + T |
| W | **320** | 38 | 31 | 6 |
| R | 1 | **368** | 2 | 32 |
| L | 0 | 1 | **359** | 21 |
| A + T | 1 | 18 | 23 | **745** |

Table 4.4: Confusion matrix for the neural network classification for the five-class problem.

| True class | Assigned class | | | | |
|---|---|---|---|---|---|
| | W | R | L | A | T |
| W | **352** | 29 | 10 | 2 | 2 |
| R | 6 | **374** | 1 | 9 | 17 |
| L | 10 | 2 | **353** | 10 | 7 |
| A | 0 | 6 | 8 | **384** | 48 |
| T | 1 | 16 | 19 | 64 | **235** |

### 4.4.4 Two-stage classifier

The objective here is to perform a "simple" classification task using a $K$-NN classifier and then use a bank of two-class neural network classifiers to handle more subtle discriminations. The first stage uses the $K$-nearest neighbor ($K = 10$) classifier to yield the two most probable classes. We observed that 85.4% of the time, the class with the maximum vote among the $K$ nearest neighbors is the correct class and 12.6% of the time, the class with the second highest vote is the correct class. In other words, the $K$-nearest neighbor classifier yields the top two classes with an accuracy of 98%. This result itself can be used to accurately classify fingerprints into two out of the five classes. Each fingerprint will have an entry in two of the five

Table 4.5: Confusion matrix for the neural network classification for the four-class problem.

| True class | Assigned class | | | |
|---|---|---|---|---|
| | W | R | L | A + T |
| W | **352** | 29 | 10 | 4 |
| R | 6 | **374** | 1 | 26 |
| L | 10 | 2 | **353** | 17 |
| A + T | 1 | 22 | 27 | **731** |

partitions of the database and the matching is required to be performed only in the corresponding two partitions of the database. The second classification stage uses 10 different neural networks for 10 different pairwise classifications of five classes. These neural networks have 192 input neurons, $20 - 40$ hidden neurons in one hidden layer, and 2 output neurons. Each neural network is trained using the patterns from only the two corresponding classes in the training set. For example, the neural network which distinguishes between $R$ and $W$ is trained using only the patterns labeled $R$ and $W$ in the training set.

Table 4.6: Confusion matrix for the two-stage classification for the five-class problem.

| True class | Assigned class | | | | |
|---|---|---|---|---|---|
| | W | R | L | A | T |
| W | **366** | 16 | 8 | 4 | 1 |
| R | 3 | **372** | 1 | 8 | 17 |
| L | 6 | 0 | **364** | 6 | 7 |
| A | 2 | 1 | 3 | **405** | 39 |
| T | 0 | 6 | 14 | 55 | **261** |

This two-stage classifier yields an accuracy of 90% for the five-class classification task and an accuracy of 94.8% is achieved for the four-class classification task. The confusion matrix for the two-stage classifier for the five-class and four-class classifi-

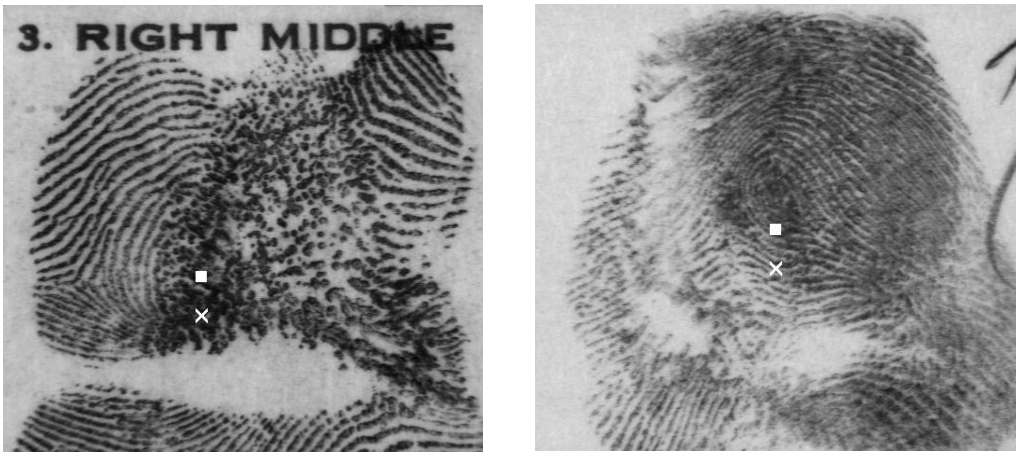
Table 4.7: Confusion matrix for the two-stage classification for the four-class problem.

| True class | Assigned class | | | |
|---|---|---|---|---|
| | W | R | L | A + T |
| W | **366** | 16 | 8 | 5 |
| R | 3 | **372** | 1 | 25 |
| L | 6 | 0 | **364** | 13 |
| A + T | 2 | 7 | 17 | **760** |

(a) Arch (b) Left Loop

Figure 4.11: Poor quality images which were correctly classified.

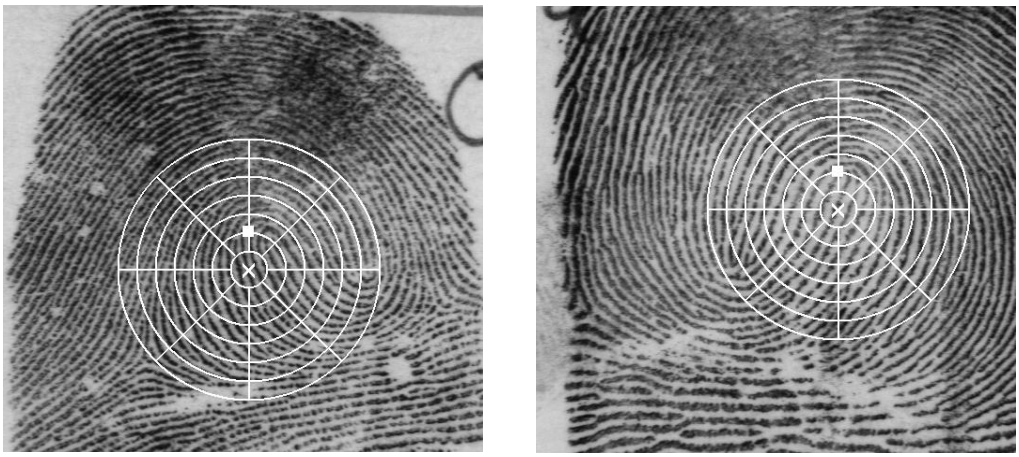
cations are shown in Tables 4.6 and 4.7 respectively. These classification accuracies do not take into account the prior class probabilities. The equal number of samples for each class in the NIST-4 database provides a relatively larger number of samples of the rare classes (arch and tented arch). However, in an operational system, the number of fingerprints for a class will be proportional to the natural distribution of fingerprints. We can estimate the performance of our two-stage fingerprint classifier on a naturally distributed database from the confusion matrices in Tables 4.6 and 4.7 by multiplying the error rate for each class with its prior probability. The estimated classification accuracy on a naturally distributed database is 93.0% for the




(a) Whorl (b) Right Loop

Figure 4.12: Poor quality images which were misclassified as arch.

five-class problem and 93.9% for the four-class problem. Although our classifier is robust to noise and is able to correctly classify most of the poor quality fingerprints in the NIST-4 database (Figure 4.11), it fails on some very bad quality fingerprint images where no ridge information is present in the central part of the fingerprint (Figure 4.12). In poor quality fingerprints it is very difficult to detect the reference point correctly (Figure 4.9 (b)). Our classifier also fails to correctly classify *twin loop* images which are labeled as *whorl* in the NIST-4 database. For these images, our reference point location algorithm picks up the upper core and on considering that as the center, the image looks like a loop in the region of interest which leads to a misclassification of $W$ as $L$ or $R$. See Figures 4.13 for these misclassifications. About 3% of the errors result from *loop-arch* misclassification because of the subtle difference between *loop* and *arch* types (see Figure 4.14(a)). The $A$-$T$ misclassification accounts for about 5% of the errors. An example of this type of confusion is shown in Figure 4.14(b).

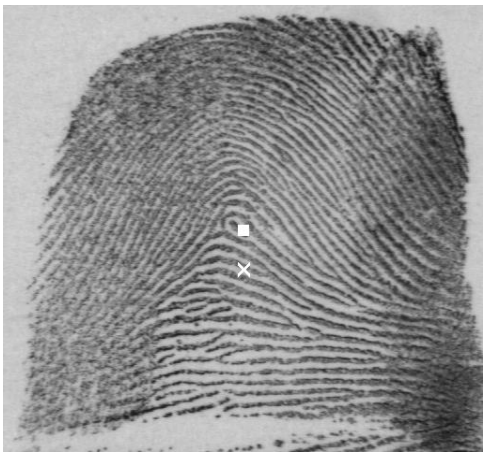


(a) Whorl as Right Loop (b) Whorl as Left Loop

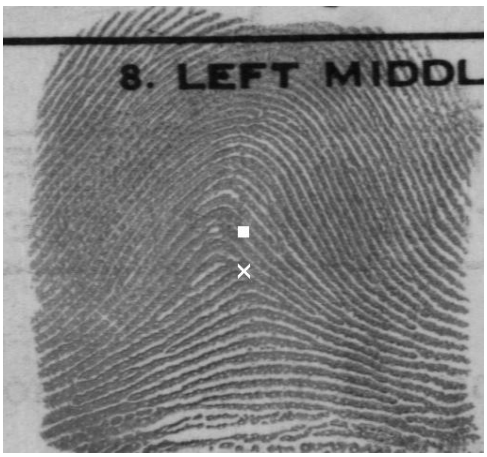
Figure 4.13: Misclassification of whorl (twin loop) as (a) right loop (b) left loop.

### 4.4.5 Reject option

Classification accuracies can be further increased by incorporating a reject option. We use the $(K, K')$-nearest neighbor classifier [145] for rejection and the proposed two-stage classifier for classification. If the number of training samples from the majority class among the $K$ nearest neighbors of a test pattern is less than $K'$ ($K' < K$), we reject the test pattern and do not attempt to classify it. Most of the rejected images using this scheme are of poor quality (Figures 4.15 (a) and (b)). Other rejected images are those images which "appear" to belong to different classes. For example, for the fingerprint image shown in Figure 4.15 (c), 3 of its nearest neighbors belong to class $R$, 3 to class $A$ and 4 to class $T$. By rejecting 19.5% of the images for the five-class problem, the classification accuracy can be increased to 93.5% and for the four-class classification problem, the accuracy can be increased to 96.6% (Table 4.8).



(a) Right loop




(b) Arch

Figure 4.14: Examples of arch-loop misclassifications; (a) a right loop misclassified as an arch; (b) an arch misclassified as a tented arch.

Table 4.8: Error-reject tradeoff.

| Classifier | (10,0)-NN (%) | (10,5)-NN (%) | (10,6)-NN (%) | (10,7)-NN (%) |
|---|---|---|---|---|
| Rejection rate | 1.8 | 8.5 | 19.5 | 32.5 |
| 5-class error | 10 | 8.8 | 6.5 | 4 |
| 4-class error | 5.2 | 4.5 | 3.4 | 2.2 |

### 4.4.6 Support vector machine classifier

For comparison purposes, we also used a support vector machine (SVM) classifier for our fingerprint classification problem. Support vector machines are kernel-based classifiers that have gained a significant popularity in recent years due to their superior performance demonstrated on a number of practical classification applications. The SVM classifiers are binary classifiers which seek that hyperplane as the decision boundary which maximizes the margin between the two classes. The parameters of a support vector machine classifier are the type of kernel, kernel parameters, and a con-


(a)

(b)

(c)

Figure 4.15: Examples of images rejected by $(10, 5)$-NN classifier.

Table 4.9: A comparison of various fingerprint classification algorithms on the NIST 4 database.

| Algorithm | Year | 5-class accuracy % (reject rate %) | 4-class accuracy % (reject rate %) |
|---|---|---|---|
| Wilson et al. [43] | 1993 | 90.2 (10) | NA |
| Karu et al. [97] | 1996 | 85.5 | 91.1 |
| Hong et al. [105] | 1999 | 87.2 | 92.3 |
| Proposed | 1999 | 90.0 (1.8) | 94.8 (1.8) |
| Cappelli et al. [141] | 1999 | 92.2 | 94.5 |

stant $c$ that controls the trade-off between the training error and the margin. We used the SVM Torch package [144] for our fingerprint classification problem. The best accuracy of 86.1% for the five-class classification task was achieved when a Gaussian kernel of standard deviation 10 and $c = 100$ was used. For a four-class problem, an accuracy of 91.8% is achieved using the same parameters. An $n$-class classification problem is solved by considering $n$ one-against-the-others support vector machine classifiers. The number of support vectors generated in the five-class classification task for the whorl-against-the others classifier was 239, the left loop-against-the-others classifier was 325, the right loop-against-the-others was 380, the arch-against-the-others was 427, and the tented arch-against-the-others was 638. Thus the total number of support vectors used by the multi-class SVM was $2,009$. Consequently, the SVM classifier is slower than the $K$-nearest neighbor classifier while providing no significant improvement in classification accuracy.

### 4.4.7 Consistency results

In the fingerprint classification task, another metric of performance evaluation is the classifier consistency. The purpose of the fingerprint classification task is to index the fingerprint database such that the input fingerprint needs to be compared only with a subset of the database. Suppose a fingerprint (let us say, of type arch) is "wrongly" classified (let us say, to type left loop) during the indexing. However, if the input fingerprint is another impression of the same finger, and is again misclassified as the same category (left loop), the indexing scheme would still be effective. The best consistency results for 967 pairs of fingerprints in the test set was achieved using a 16-nearest neighbor classifier as 82.6% for the five-class classification and 89.8% for the four-class classification. The classification results stated in Table 4.8 made use of the multiple labels of the fingerprint images in the NIST-4 database. However, if we strictly consider only the first label of the fingerprint images in the NIST-4 database for a fair comparison, the $K$-nearest neighbor fingerprint classifier gives a five-class classification accuracy of 79.8% and four-class classification accuracy of 88.3%. Thus, the consistency result is 2.8% ($82.6\% - 79.8\%$) better than the accuracy result for the five-class problem and 1.5% ($89.8\% - 88.3\%$) better for the four-class problem.

### 4.4.8 Defining New Classes

The five fingerprint classes, i.e., whorl, left loop, right loop, arch, and tented arch, used in this chapter are based on the Henry system of classification which has been in use for more than one hundred years. These classes used in the forensic domain

may not be the best separable categories in our filterbank-based feature space. It is possible to define new classes such that the fingerprints belonging to different classes are compact and well separated in the feature space. We first assumed that the "clusters" formed by the fingerprint patters in the FingerCode-space are essentially spherical. As a result, we used a standard $k$-means clustering algorithm that uses Euclidean distance metric on the training data to detect clusters in the feature space. The clusters thus detected do not have any physical meaning in terms of fingerprint patterns and define non-intuitive fingerprint categories. Since the $k$-means clustering algorithm depends on the initialization of cluster centers, we performed multiple (20) runs of the $k$-means algorithm with different initializations and chose the clustering with the minimum squared error. The best consistency results on the 967 pairs of test images were 73.8% (using a 7-nearest neighbor classifier) when five clusters were defined and 82.2% (using a 12-nearest neighbor classifier) when four clusters were defined. On changing the distance metric for the $k$-means algorithm from Euclidian distance to Mahalanobis distance [145], the $k$-means algorithm seeks hyper-elliptical clusters instead of spherical cluster. We achieve sightly higher consistency results of 76.2% for the four-class problem and 85.2% for the five-class problem by using the Mahalanobis distance. This suggests that the shape of the clusters formed in the FingerCode feature space is closer to elliptical than spherical. However, the consistency results when the classes were defined using a clustering of the data are inferior as compared to the consistency results when the classes were defined by a fingerprint expert. This implies that the fingerprints do not form well defined clusters in the FingerCode feature space and an exclusive classification of fingerprints has

limitations because of the inherent overlap between the fingerprint classes. Therefore, a continuous classification of fingerprints should be explored. A successful continuous fingerprint classifier is developed by Lumini et al. in [22].

### 4.4.9 Dimensionality Reduction Using PCA

The training and the test sets contain about $2,000$ samples each while our feature vector is 192-dimensional. It is desirable to have a large number of representative samples per class (e.g., ten times) with respect to the feature dimension for good generalization of a practical classification system [145]. Since the collection of a large number of representative samples is expensive, we used the principal component analysis (KL transform) to reduce the dimensionality of the feature vector and used a K-nearest neighbor algorithm classifier. While an accuracy of 85.4% was achieved with this KL-KNN classifier when all the 192 features were used, we achieved an accuracy of 85.1% when only 96 features were used (8-nearest neighbor classifier), 84.3% when 72 features were used (10-nearest neighbor classifier), and 83% when 48 features were used (12-nearest neighbor classifier). Thus, with a slight degradation in performance, we were able to reduce the feature vector size to 1/4th of its original value. A similar behavior was observed in the classifier consistency results as well. The consistency was 81.1% for 96 features, 81.0% for 72 features, and 79.2% for 48 features.

Cappelli et al. [141] used a Multi-space KL transform for feature reduction. The central idea of this approach is to find one or more KL subspaces for each class that

are well-suited in representing the fingerprints in that class. They used a fixed number of subspaces for each class. The selection of the number of subspaces for a class was ad-hoc and was based on the authors' perception of complexity of that class (arch, left loop, right loop, whorl, and tented arch, were assigned 1, 2, 2, 3, and 1 subspaces, respectively). An accuracy of over 99% with 20% reject rate was reported on the naturally distributed NIST-14 database using a combination of six classifiers including the Multi-KL-KNN classifier which was the best individual classifier. The accuracy on the NIST-4 database that contains equal number of fingerprint images from the five classes was not reported and is expected to be inferior due to inclusion of large number of more difficult arch and tented arch type fingerprint images. We used a similar idea to develop a Multi-KL-KNN classifier based on the filterbank representation. We used an equal number of subspaces for all the five classes (one for each class) because the complexity of each class in not known apriori in the filterbank representation. We were able to achieve a five-class classification accuracy of 85.1% when only 96 features were used for each class, an accuracy of 84.9% was achieved when 72 features were used, and an accuracy of 83.2% was achieved when only 48 features were used. This shows that Multi-KL-KNN classifier performs marginally better than the KL-KNN classifier but not as good as without dimensionality reduction.

### 4.4.10 Dimensionality Reduction Using Feature Clustering

Although the feature dimension reduction using principal component analysis is useful as the final classifier is based on fewer features and as a consequence, is faster, the

feature extraction time is not reduced. All the 192 features are first extracted from the fingerprint images and then the feature vector is reduced by projecting it to the new space. In order that the feature extraction time is reduced, we need to "select" a subset of features while maintaining the classification accuracy. For this purpose, we used a standard $k$-means clustering algorithm to cluster the features into 96 and 48 clusters, respectively. Due to very few number of samples in each cluster, the clustering results are not used directly for feature reduction. We observe that the corresponding feature values for the same location but different orientation cluster together. This means that there is some redundancy in the different directions (4 in our case) used for the Gabor filters during the feature extraction. However, each direction yields some extra information such that the classification accuracy increases by using more directions. The increase in the classification accuracy that results from using more number of orientation specific filters result in increased computation time for feature extraction. Depending on the application, the tradeoff between accuracy and time can be selected. For example, using a $K$-nearest neighbor classifier, an accuracy of 85.4% is achieved by using 4 directions, an accuracy of 82.0% is achieved when 2 directions are used and an accuracy of 65% is achieved when only one direction is used. A similar behavior was observed in the classification consistency results as well. The consistency was 82.6% for four directions, 79.1% for two directions, and 61.9% when only one direction was used.

## 4.5 Summary

We have developed a fingerprint classification algorithm that uses the filterbank-based representation and outputs an accuracy comparable to the state-of-the-art algorithms reported in the literature on the NIST-4 database. Our feature vector, called FingerCode, captures the fingerprint class information and is robust to noise which is reflected in the high classification accuracy. We have tested our algorithm on the NIST-4 database and a very good performance has been achieved (90% for the five-class classification problem and 94.8% for the four-class classification problem with 1.8% rejection during the feature extraction phase). However, this algorithm suffers from the requirement that the region of interest be correctly located, requiring the accurate detection of reference point in the fingerprint image. Our system takes about 3 seconds on a Sun Ultra-10 machine to classify one fingerprint. Since image decomposition (filtering) steps account for 90% of the total compute time, special purpose hardware for convolution can significantly decrease the overall time for classification. Most of the work in fingerprint classification has concentrated on features (e.g., location of singular points, orientation field) that the forensic scientists have used for a long time. These classifiers perform discrete classification of fingerprint images into one of the predetermined classes. Since there exits a continuum of fingerprint patterns between these discrete predetermined classes, the automatic systems based on simple features such as singular points or orientation field will have a limited performance irrespective of the location and shape of the decision boundary when performing discrete classification. By attempting to design features which are

parameterized, rich and completely data driven, such as the ones proposed in this thesis, we can apply advanced pattern recognition and clustering techniques instead of simple hand-crafted rules to gain performance improvement. We believe that the FBI requirement of 1% error with 20% reject rate is very challenging to meet. The algorithms that have reported a performance close to or surpassing this requirement [44, 141] have reported their results on a naturally distributed database and have thus taken the advantage of the fact that the less frequently occurring classes are more difficult to classify. We have shown that the simple variance-based features proposed in this thesis work quite well. However, we expect that better performance can be achieved by extracting richer, more discriminatory features from the filtered images in the feature extraction algorithm.

# Chapter 5

# Fingerprint Matching

The distinctiveness of a fingerprint can be determined by the overall pattern of ridges and valleys as well as the local ridge anomalies (minutiae points). Although the ridges possess the discriminatory information, designing a reliable automatic fingerprint matching algorithm is very challenging due to the nonlinear deformation and noise in fingerprint images (see Figure 3.2).

The existing popular fingerprint matching techniques can be broadly classified into two categories: (a) minutiae-based and (b) correlation-based. The minutiae-based techniques typically match the two minutiae sets from two fingerprints by first aligning the two sets and then counting the number of minutiae that match. A typical minutiae extraction technique performs the following sequential operations on the fingerprint image: (*i*) fingerprint image enhancement, (*ii*) binarization (segmentation into ridges and valleys), (*iii*) thinning, and (*iv*) minutiae detection. Several commercial [112] and academic [131, 11] algorithms follow these sequential steps for minutiae detection. Alternative techniques for minutiae detection directly operate

on the gray scale fingerprint image itself and detect minutiae by adaptively tracing the gray scale ridges in the fingerprint images [56, 181]. The alignment between the input and the template fingerprints can be obtained using one or more of the fingerprint features. For example, an alignment can be achieved based on the orientation field of the fingerprints, the location of singular points such as the core and the delta [95], ridges [11], inexact graph-matching on the minutiae graphs [5], Hough transform [131], point patterns [128]), etc. The number of matched minutiae in certain tolerances is typically normalized by the total number of minutiae in the two sets to account for the falsely detected and missed minutiae during the feature extraction. One of the main difficulties in the minutiae-based approach is that it is very difficult to reliably extract minutiae in a poor quality fingerprint image. A number of image enhancement techniques can be used to improve the quality of the fingerprint image prior to minutiae extraction (e.g., [108]).

Correlation-based techniques match the global pattern of ridges and furrows to see if the ridges align. The simplest technique is to align the two fingerprint images and subtract the input from the template to see if the ridges correspond. However, such a simplistic approach suffers from many problems including the errors in estimation of alignment, non-linear deformation in fingerprint images, and noise. An auto-correlation technique has been proposed by Sibbald [31] that computes the correlation between the input and the template at fixed translation and rotation increments. If the correlation exceeds a certain threshold, the two fingerprints are declared to originate from the same finger. A variant of the correlation technique is to perform the correlation in the frequency domain instead of the spatial domain by performing

a two-dimensional fast Fourier transform (FFT) on both the input and the template fingerprints. The sum of the pixel-to-pixel multiplication of the two frequency domain representations of the fingerprint images is then compared to a threshold to make a decision. One of the advantages of performing correlation in the frequency domain is that the frequency representations of the fingerprints are translation invariant. One of the major disadvantages, however, is the extra computation time required to convert the spatial image to a frequency representation. The frequency domain correlation matching can also be performed optically [137, 38, 73]. The input and the template fingerprints are projected via laser light through a lens to produce their Fourier transform and their superposition leads to a correlation peak whose magnitude is high for the matching pair and low otherwise. The main advantage of performing optical correlation is the speed; the main disadvantage is that optical processors have very limited versatility (programmability) (cf. [112]). A modification of the spatial correlation-based techniques is to divide the fingerprint images into grids and determine the correlation in each sector instead of the whole image [61, 103]. The correlation-based technique overcomes some of the limitations of minutiae-based approach. For example, the minutiae extraction algorithm detects a large number of spurious minutiae and misses genuine minutiae in very noisy fingerprint images. Correlation-based techniques are less sensitive to the noise in fingerprint images but have problems of their own. For example, correlation-based techniques are more sensitive to an error in estimation of the alignment between the two fingerprints. Also, the correlation-based techniques cannot easily deal with the non-linear deformation present in the fingerprint images. Additionally, the correlation-based techniques typi-

cally have larger template size. See Table 5.1 for a comparison of different fingerprint matching algorithms.

Table 5.1: Fingerprint matcher literature survey. The fingerprint matching algorithms are classified based on the alignment assumed between the template and the input fingerprint features. The rotation is denoted by $R$, the translation is denoted by $T$, and the scale is denoted by $S$.

| Author (Year) | Alignment | Features Used |
|---|---|---|
| Kovacs-Vajna [186] (2000) | nonlinear | minutiae and its $16 \times 16$ grayscale neighborhood |
| Jiang et al. [182] (2000) | R + T + S | minutiae |
| Almansa and Cohen [1] (2000) | nonlinear | minutiae |
| Jain et al. [19] (2000) | R + T | texture features |
| O'Gorman [112] (1999) | R + T in local regions | minutiae |
| Jain et al. [11] (1997) | R + T + nonlinear | thin ridges, minutiae |
| Sibbald [31] (1997) | R + T | grayscale intensity |
| Ratha et al. [131] (1996) | R + T + S | minutiae |
| Maio et al. [58] (1995) | R + T | minutiae, core, delta |
| Coetzee and Botha [103] (1993) | R + T | minutiae and frequency-domain features |
| Marsh and Petty [137] (1991) | R + T | grayscale intensity |
| Driscoll et al. [61] (1991) | R + T | grayscale intensity |

The filterbank-based representation described in Chapter 3 does not fall either into the minutiae-based or the correlation-based matching categories. The proposed technique is a feature-based technique that captures both the local and the global details in a fingerprint as a compact fixed length feature vector (FingerCode). The fingerprint matching is based on the Euclidean distance between the two corresponding FingerCodes and hence is extremely fast. We are able to achieve a verification accuracy superior to the results of a typical state-of-the-art minutiae-based algorithm [11] in terms of equal error rates (see Table 5.3) and only marginally inferior at very

low false accept rates on two different databases. Finally, we show that the matching performance can be improved by combining the decisions of the matchers based on complementary (minutiae-based and filter-based) fingerprint information.

## 5.1 Introduction

It is desirable to explore representation schemes which combine global and local information in a fingerprint. Our novel, relatively short, fixed length code representation for the fingerprints, called FingerCode is suitable for matching as well as storage on a smartcard. The matching reduces to finding the Euclidean distance between these FingerCodes and hence the matching is very fast and the representation is amenable to indexing.

## 5.2 Feature Extraction

We have used the proposed filterbank-based representation described in Chapter 3 with the values of the parameters described below. In our initial experiments with MSU_DBI database (image size = $508 \times 480$ pixels, scanned at 500 *dpi*), we considered five concentric bands ($B = 5$) for feature extraction. Each band is 20-pixels wide ($b = 20$), and segmented into sixteen sectors ($k = 16$) (Figure 3.9). Thus, we have a total of $16 \times 5 = 80$ sectors ($S_0$ through $S_{79}$) and the region of interest is a circle of radius 120 pixels, centered at the reference point. Eighty features for each of the eight filtered images provide a total of 640 ($80 \times 8$) features per fingerprint image.

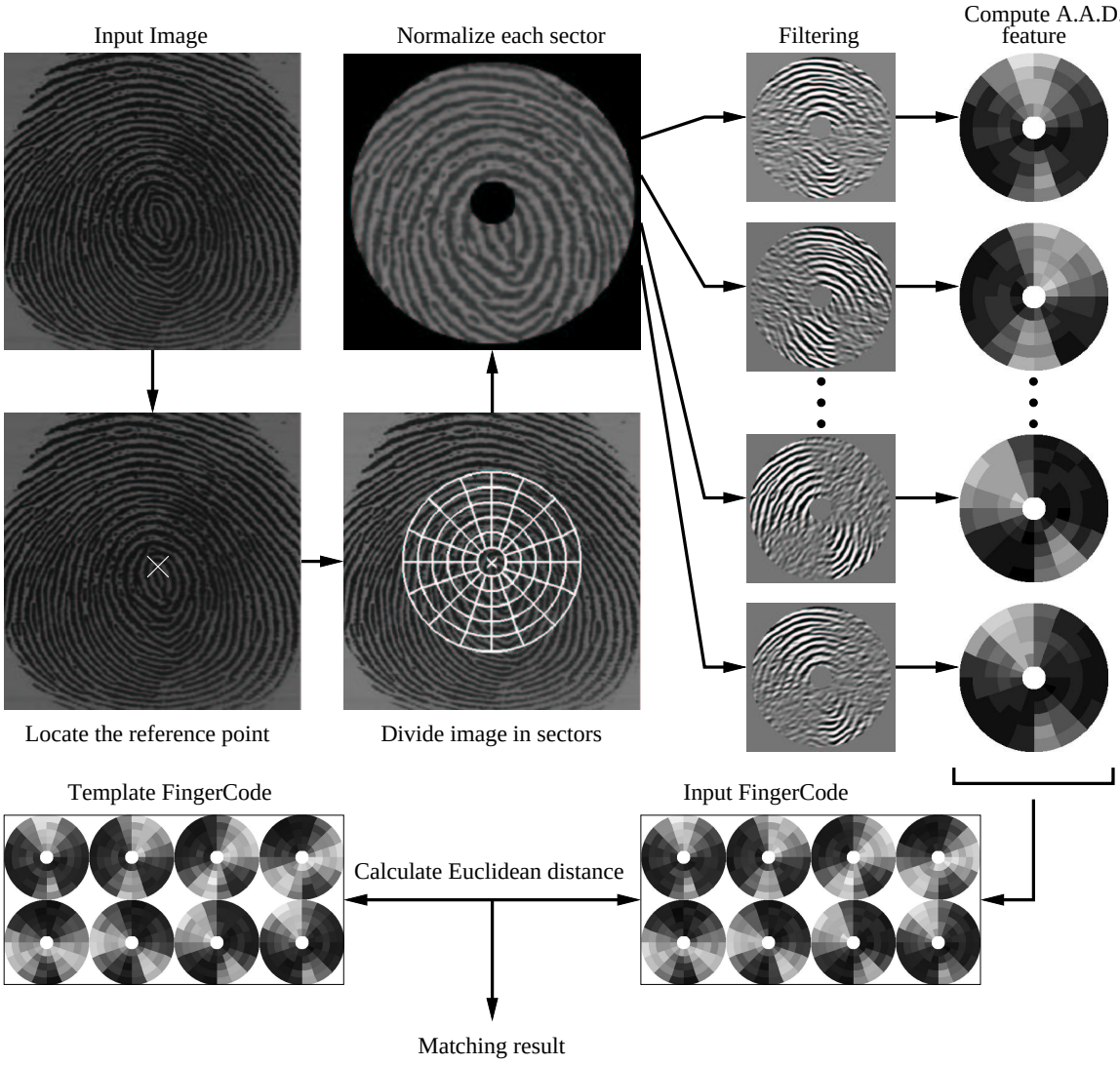



Figure 5.1: System diagram of our fingerprint authentication system.

Each feature can be quantized into 256 values and requires 1 byte of storage, so the entire feature vector requires only 640 bytes of storage. Note that these parameters of the tessellation depend upon the image resolution and size. In our second experiment with NIST-9 database (image size = $832 \times 768$ pixels, scanned at 500 *dpi*), we used 7 concentric bands ($B = 7$), $b = 20$, and $k = 16$, giving us an 896 byte FingerCode.

The 640-dimensional feature vectors (FingerCodes) for nine different impressions of the same finger are shown as gray level images with eight disks, each disk corresponding to one filtered image in Figure 5.2. The gray level in a sector in a disk represents the feature value for that sector in the corresponding filtered image. The nine fingerprint images of the different impressions of the same finger differ from each other in translation, rotation, non-linear deformation, image intensities in different part of the fingerprint, and noise. A simple correlation-based technique that subtracts the input image from the template is unlikely to succeed due to the large intra-class variation in different impressions of the same finger. The tessellation scheme of the proposed filterbank-based algorithm is able to handle small errors in the location of the reference point for translation invariance of the representation. One can see that the representations for the nine impressions of the same finger visually look very similar. However, the feature values in different impressions are not exactly the same. This difference results from the image rotation and non-linear deformation. However, the gray scale intensity-based feature (variance) is able to handle small rotation and non-linear deformation in each sector. The Euclidean distance between these nine FingerCodes ranges from 10 and 40 (normalized to a scale of 0-100) which quantifies the typical intra-class variability in a single user. The Euclidean distance between the

FingerCodes from different fingers range from 30 to 100 on the same scale quantifying the typical inter-class variability in the representation.

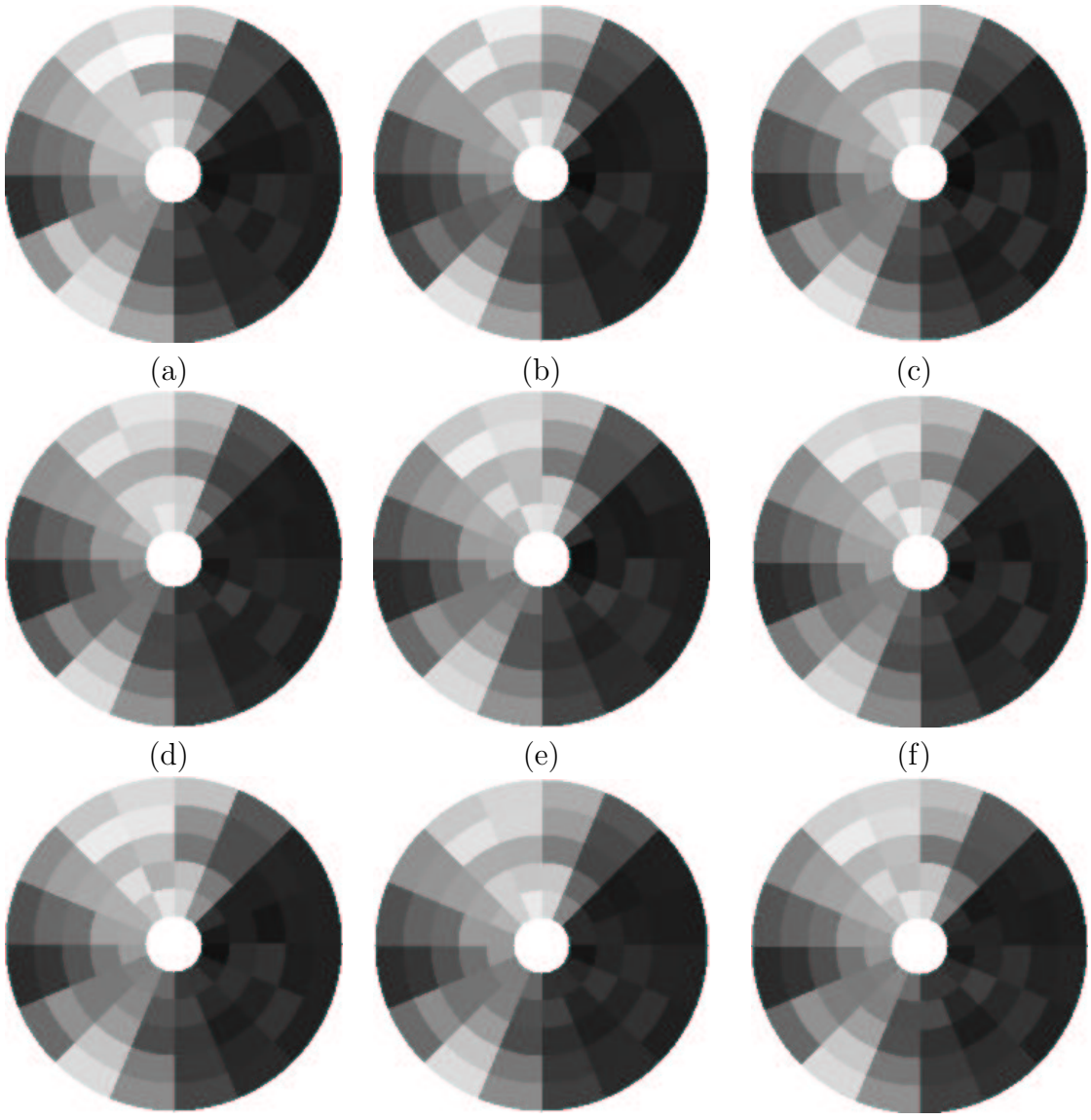



Figure 5.2: Examples of 640-dimensional feature vectors corresponding to nine different impressions of the same finger.

## 5.3 Matching

Fingerprint matching is based on finding the Euclidean distance between the corresponding FingerCodes. The translation invariance in the FingerCode is established by identifying the reference point. However, FingerCodes are not rotationally invariant. The approximate rotation invariance is achieved by cyclically rotating the features in the FingerCode itself. A single-step cyclic rotation of the features in the FingerCode described by Eqs. (5.1)-(5.3) corresponds to a feature vector which would be obtained if the image was rotated by 22.5°. A rotation by $R$ steps corresponds to a $R \times 22.5°$ rotation of the image. A positive rotation implies counterclockwise rotation while a negative rotation implies clockwise rotation. See Figure 5.3 for an illustration. The FingerCode obtained after $R$ steps of rotation is given by

$$V_{i\theta}^{R} = V_{i'\theta'}, \tag{5.1}$$

$$i' = (i + k - R) \; mod \; k + (i \; div \; k) \times k, \tag{5.2}$$

$$\theta' = (\theta + 180° + 22.5° \times (-R)) \; mod \; 180°, \tag{5.3}$$

where $V_{i\theta}^{R}$ is the rotated FingerCode, $V_{i'\theta'}$ is the original FingerCode, $k$ (= 16) is the number of sectors in a band, $i \in [0, 1, 2, ...79]$, and $\theta \in$ [0°, 22.5°, 45°, 67.5°, 90°, 112.5°, 135°, 157.5°].

For each fingerprint in the database, we store five templates corresponding to the following five rotations of the corresponding FingerCode: $V_{i\theta}^{-2}$, $V_{i\theta}^{-1}$, $V_{i\theta}^{0}$, $V_{i\theta}^{1}$, and $V_{i\theta}^{2}$. We use only five values of parameter $R$ $(-2, -1, 0, 1, 2)$ because the fingerprint

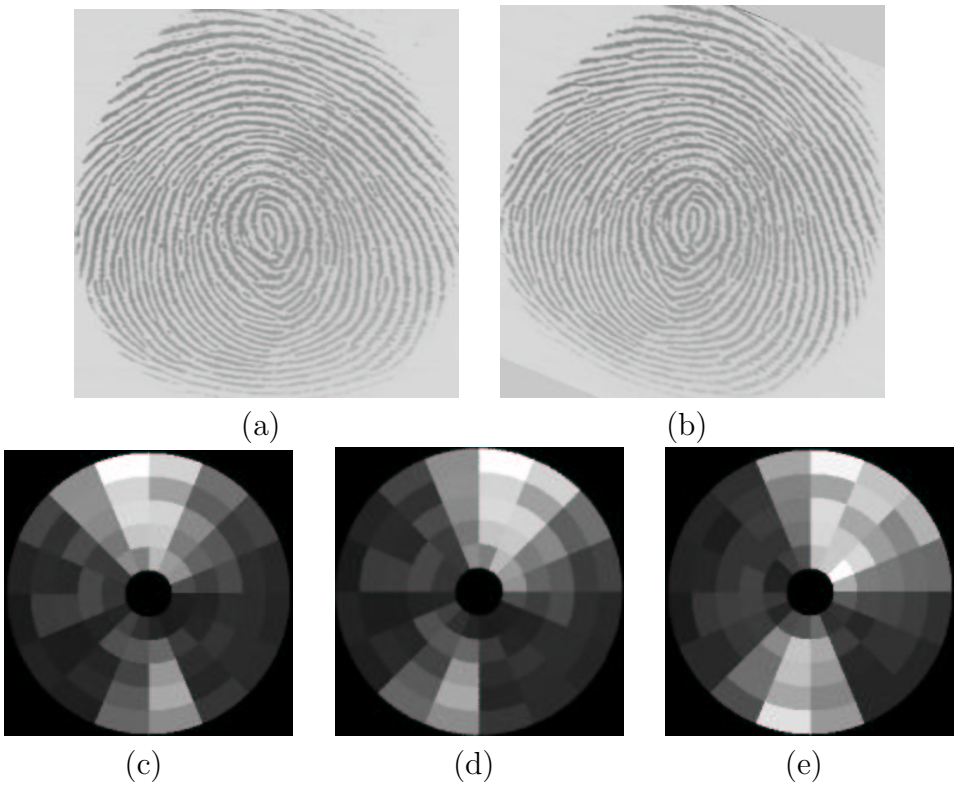



Figure 5.3: The fingerprint image in (b) is obtained by a $-22.5°$ rotation of (a). A part of the feature vector corresponding to the $0°$ Gabor filtered image extracted from (a) is shown in (c) as a gray scale image. The feature vector in (c) is rotated by $-22.5°$ ($R = -1$ in Equations (5.2) and (5.3)) and is shown in (d). (e) shows the feature vector extracted from the fingerprint image in (b). The feature vectors shown in (d) and (e) are similar illustrating that the feature vector for a $-22.5°$ rotation in the original image approximately corresponds to a unit anticlockwise cyclic rotation of the feature vector.

images in both the MSU_DBI and NIST-9 databases do not have more than $\pm 45°$ rotation. For databases that have more rotation in the fingerprint images, a higher range for the parameter $R$ may be used. The input FingerCode is matched with the five templates stored in the database to obtain five different matching scores. The minimum of these five matching scores corresponds to the best alignment of the input fingerprint with the database fingerprint. Since a single cyclic rotation of the features in the FingerCode corresponds to a rotation of $22.5°$ in the original image, we can only generate those representations of the fingerprint which are in

multiples of 22.5°. Due to the nature of the tessellation, our features are invariant to only small perturbations that are less than ±11.25°. Therefore, we generate another feature vector for each fingerprint at the time of user enrollment which corresponds to a rotation of 11.25°. The original image is rotated by an angle of 11.25° and its FingerCode is generated. Five templates corresponding to the various rotations of this FingerCode are also stored in the database. Thus, the database contains 10 templates for each fingerprint. These 10 templates correspond to all the rotations of the fingerprint image in multiples of 11.25°. This takes care of the fingerprint rotation while matching the input FingerCode with the stored templates. The final matching distance score is taken as the minimum of the ten scores obtained by matching the input FingerCode with each of the 10 templates. This minimum score corresponds to the best alignment of the two fingerprints being matched. Since the template generation for storage in the database is an off-line process and the matching process is extremely fast, the verification time still depends on the time taken to generate a single template for the test image.

## 5.4 Experimental Results

Our MSU_DBI database consists of a total of 2,672 fingerprint images from 167 subjects. A live feedback of the acquired image was provided during the data capture and the volunteers guided the subjects in placing their fingers in the center of the sensor and in an upright position. Due to this assistance provided to the subjects, most of the fingerprints were reasonably well centered. Despite the supervised image

acquisition, there is a significant intra-class deformation and up to $\pm 45^{\circ}$ deviation from the assumed vertical upright orientation in the acquired images. However, these images are of better quality than the traditional inked fingerprints (see Figure 5.4). The fingerprint images which were captured after a period of six weeks have significant nonlinear distortions due to finger pressure differences (see Figures 5.5 and 5.7(c) and (d)). This presents a challenge to all the fingerprint matching algorithms.

(a) (b)

Figure 5.4: A comparison of the quality of inked fingerprints and dab fingerprints. (a) inked fingerprint, (b) dab fingerprint.

We have also evaluated our system on $1,800$ images of the public domain database NIST-9 (Vol. 1, CD. No. 1) which contains $1,800$ fingerprint images (image size = $832 \times 768$ pixels) from 900 different fingers. The complete NIST-9 fingerprint database contains $1,350$ mated fingerprint card pairs ($13,500$ fingerprint image pairs) that approximate a natural distribution of the National Crime and Information Center fingerprint classes. The database is divided into multiple volumes. Each volume has

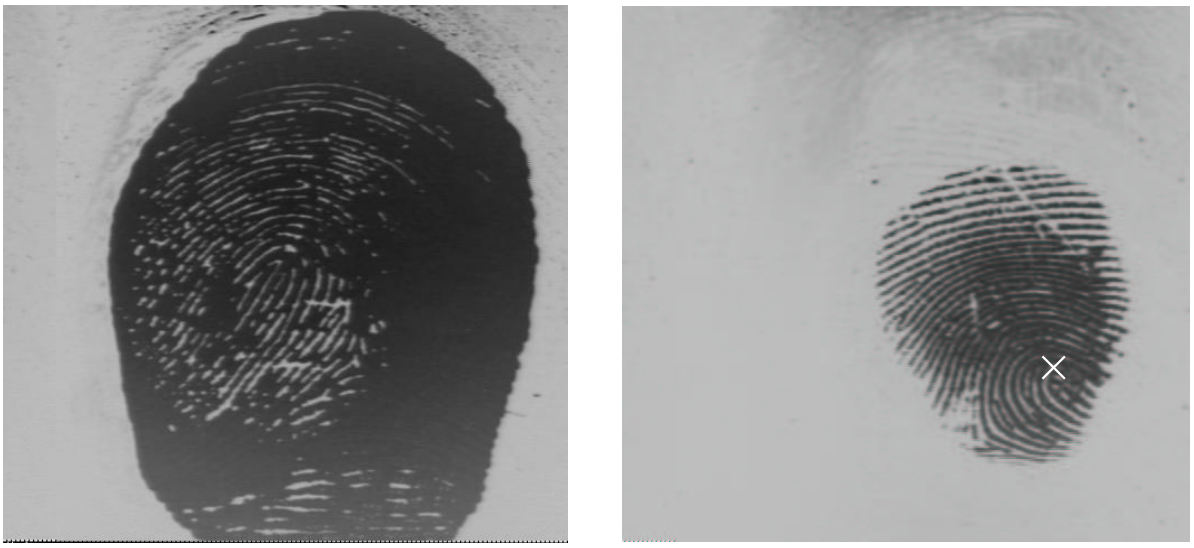
Figure 5.5: Examples of images with large deformation due to finger pressure differences in the MSU_DBI database. Fingerprint images in (b) and (d) were taken six weeks after the images in (a) and (c) were acquired, respectively.

three compact discs (CD's). Each CD contains 900 images of card type 1 and 900 images of card type 2. Fingerprints on the card type 1 were scanned using a rolled method, and fingerprints on card type 2 were scanned using a live-scan method. Matching fingerprint images in the NIST-9 database is more difficult compared to the live-scan fingerprint images because the two impressions from the same finger in the NIST-9 database are captured using different methods (rolled and live-scan) and

hence the two images of the same finger differ significantly in their ridge structures. A large number of NIST-9 images are of poorer quality and these images often contain extraneous objects like handwritten characters and other artifacts common to inked fingerprints.

One hundred images (approximately 4% of the database) were rejected from the MSU_DBI database because of the following reasons: (*i*) the reference point was located at a corner of the image and therefore an appropriate region of interest (tessellation) could not be established, (*ii*) the quality of the image was poor based on the quality index of the images. See Figure 5.6 for examples of images which were rejected. A total of 100 images (approximately 5.6% of the database) were rejected from the NIST-9 database based on the same criteria. The quality index was determined using a quality checker algorithm [147] that estimates the dryness of the finger (or smudginess of the fingerprint image) and the extent to which the surface of the finger tip is imaged. The estimate of the dryness/smudginess is based on the variance of the grayscale in the captured image.

To establish the verification accuracy of our fingerprint representation and matching approach, each fingerprint image in the database is matched with all the other fingerprints in the database. A matching is labeled correct if the matched pair is from the same finger and incorrect, otherwise. None of the genuine (correct) matching scores was zero indicating that the images from the same finger did not yield an identical FingerCode because of the rotation, distortion, and inconsistency in reference point location. For the MSU_DBI database, a total of $3,306,306$ matchings were performed. The probability distribution for genuine (correct) matches was estimated



(a) (b)

Figure 5.6: Examples of rejected images. (a) a poor quality image, (b) the reference point is (correctly) detected at a corner of the image and so an appropriate region of interest could not be established.

with $7,472$ matches and the imposter distribution was estimated with $3,298,834$ matches. Figure 5.8 (a) shows the two distributions. For the NIST-9 database, a total of $722,419$ matchings were performed and the genuine and imposter distributions were estimated with $1,640$ and $720,779$ matching scores, respectively. Figure 5.8 (b) shows the imposter and genuine distributions for the NIST-9 database. If the Euclidean distance between two FingerCodes is less than a threshold, then the decision that "the two images come from the same finger" is made, otherwise a decision that "the two images come from different fingers" is made. Different decision thresholds lead to different values of FAR and FRR (see Table 5.2).

A Receiver Operating Characteristic (ROC) curve is a plot of Genuine Acceptance Rate (1-FRR) against False Acceptance Rate for all possible system operating points (i.e., matching distance threshold) and measures the overall performance of the system. Each point on the curve corresponds to a particular decision threshold. In the

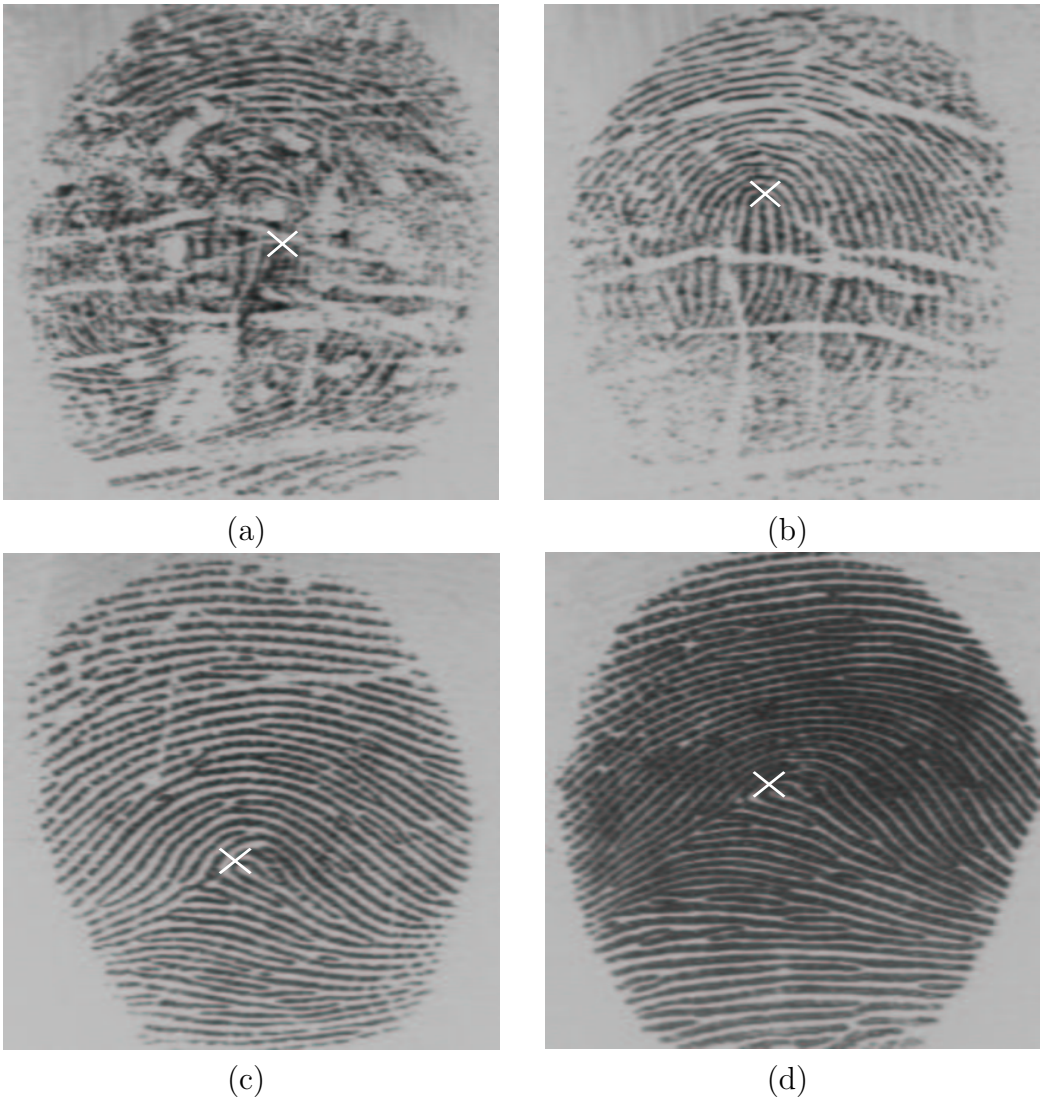


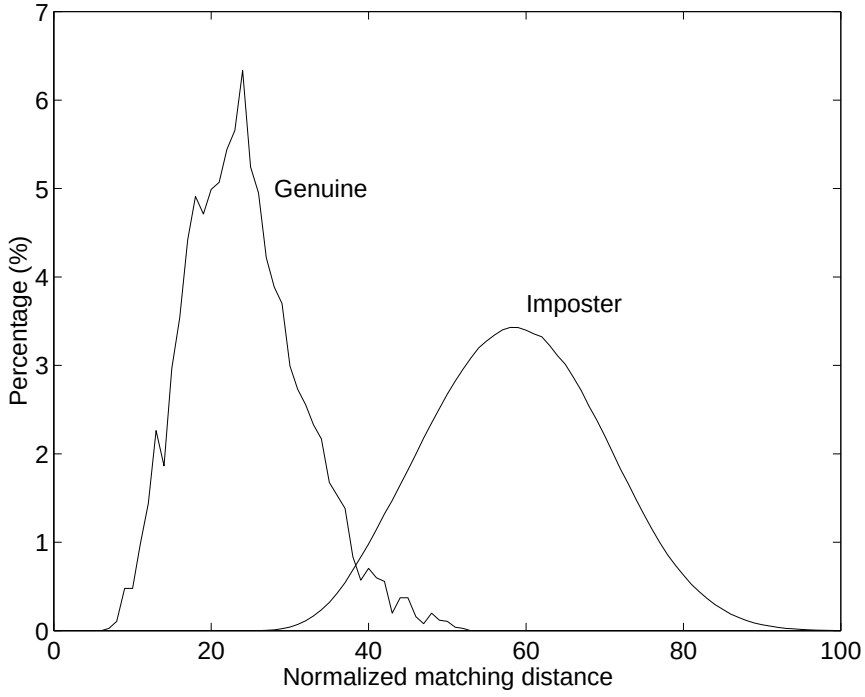

Figure 5.7: Errors in matching. Examples of fingerprint images from the same finger that were not correctly matched by our algorithm. (a) and (b) do not match because of the failure of reference point location, (c) and (d) do not match because of the change in inter ridge distances due to finger pressure difference.

Table 5.2: False acceptance and false reject rates with different threshold values for the MSU_DBI database.

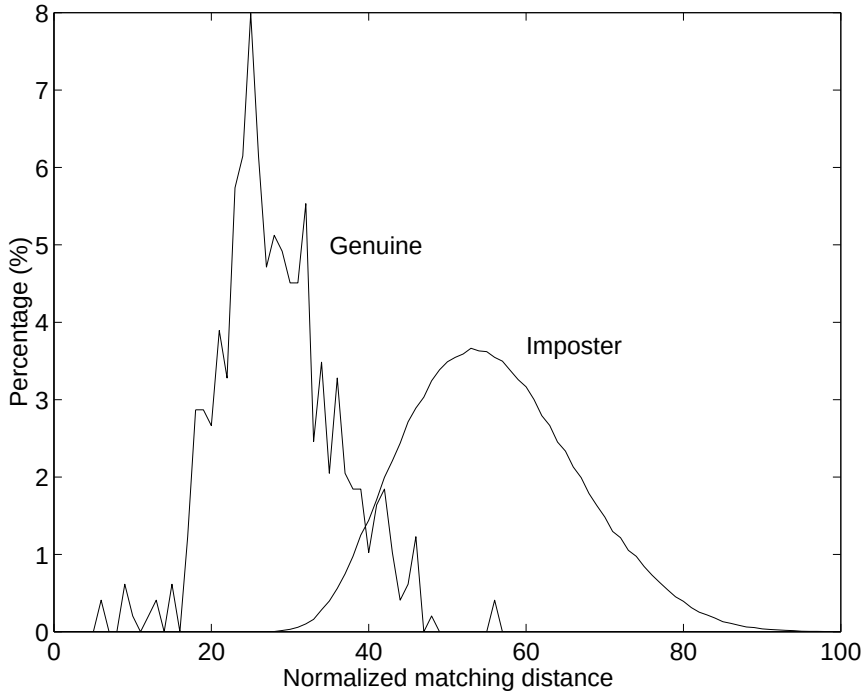
| Threshold value | False Acceptance Rate (%) | False Reject Rate (%) |
|---|---|---|
| 30 | 0.10 | 19.32 |
| 35 | 1.07 | 7.87 |
| 40 | 4.59 | 2.83 |

ideal case, both the error rates, i.e., FAR and FRR should be zero and the genuine distribution and imposter distribution should be disjoint. In such a case, the "ideal" ROC curve is a step function at the zero False Acceptance Rate. On the other extreme, if the genuine and imposter distributions are exactly the same, then the ROC is a line segment with a slope of 45° with an end point at zero False Acceptance Rate. In practice, the ROC curve behaves in between these two extremes. An Equal Error Rate (EER) is defined as that operating point where the two types of errors, FAR and FRR, are equal. Figures 5.9 (a) and (b) compare the ROCs of a state-of-the-art minutiae-based matcher [11] with our filter-based matcher on the MSU_DBI and the NIST-9 databases, respectively. The ROC curves show that our system performs better than the minutiae-based system when the system performance requirements are less demanding on FAR (FAR greater than 2%) on both the databases. A number of applications including bank's ATM machines usually have such FAR requirements. However, at very low FARs, our system performs worse than the minutiae-based approach. Our system also performs better than the minutiae-based system at the equal error rate (see Table 5.3).

Most of the false accepts in our system occur among the same "type" (class) of fingerprints; a whorl is confused with another whorl and not a loop. This confirms

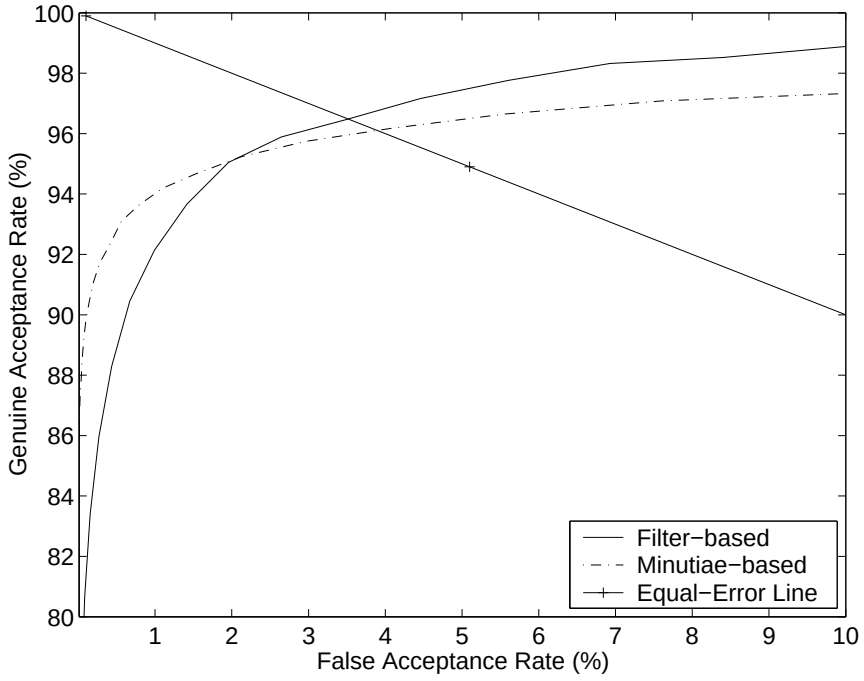



(a)

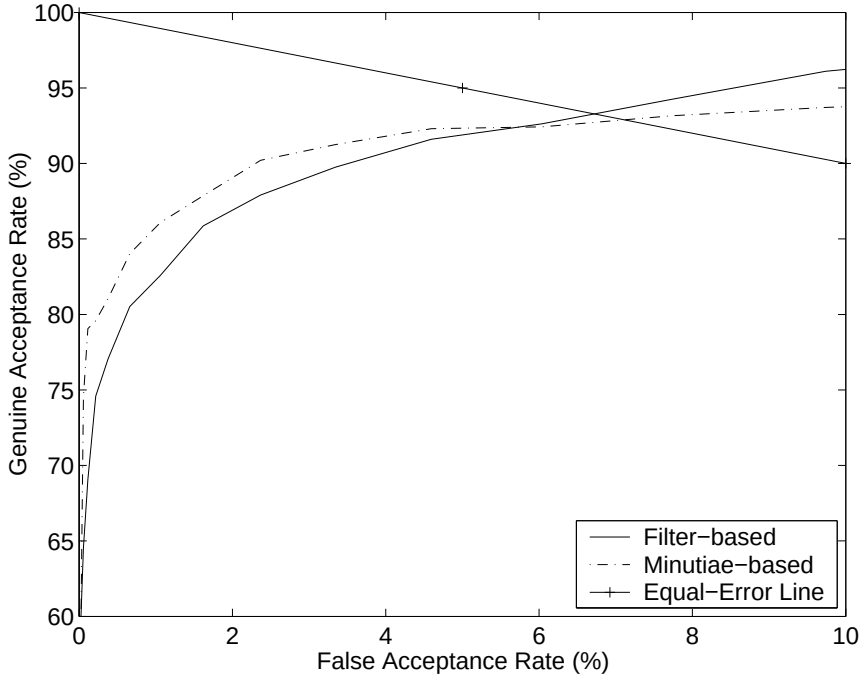



(b)

Figure 5.8: Genuine and imposter distributions for the proposed verification scheme. (a) MSU_DBI database, (b) NIST-9 (Vol. 1, CD No. 1).




(a)




(b)

Figure 5.9: Receiver Operating Characteristic (ROC) curves for two different (filterbank-based and minutiae-based) matchers. (a) MSU_DBI database, (b) NIST-9 (Vol. 1, CD No. 1). FAR and FRR are equal at all points on the Equal-Error Line. Thus, the point of crossing of ROC with this line denotes the equal error rate on the ROC.

that the proposed approach does capture the global information as well as the local information and hence is suitable for indexing as shown in Chapter 4. However, this is a shortcoming in terms of verification. The imposter distributions in Figure 5.8 are wider than the typical imposter distribution in a minutiae-based approach. This also suggests that FingerCodes capture more global information; at a global information level, there is more similarity in fingerprints from different fingers. Since FingerCodes also capture local information, there is more variation in imposter scores which results in a wider imposter distribution than the minutiae-based technique. The genuine distribution for the minutiae-based approach is typically very wide because the matching score depends heavily on the quality of the fingerprint image. The filterbank-based approach, on the other hand, has a relatively narrower genuine distribution due to its superior ability to deal with noise in the fingerprint image. As a result of this difference in the characteristics of the genuine and the imposter distributions, when the threshold is changed from a value corresponding to a very low FAR to high FARs, the FRR for the minutiae-based approach drops very rapidly and then stabilizes. On the other hand, the FRR for the filterbank-based approach drops slowly but steadily with increasing FAR leading to a crossover of its ROC with the minutiae-based ROC (see Figures 5.9(a) and (b)). This implies that the filterbank-based approach is superior to the minutiae-based approach at high FARs due to its ability to gracefully deal with large amount of noise in the fingerprint images. The filterbank-based approach is inferior to the minutiae-based approach at low FARs because it is capturing more global information and is not able to distinguish between fingerprints that have a very similar global structure. This suggests that the FingerCode representation captures

discriminatory information that is complementary to the information used by popular minutiae-based fingerprint matchers. An added advantage of such independent knowledge is that a combination of the two approaches, i.e., filterbank-based and minutiae-based, can significantly improve the overall performance of the verification system. This will be further discussed in Chapter 6.

Table 5.3: Comparison of the equal error rates (ERR) of the proposed filterbank-based technique with a state-of-the-art minutiae-based technique on two different databases.

| Database | Minutiae-based (%) (reject rate (%)) | Filterbank-based (%) (reject rate (%)) |
|---|---|---|
| MSU_DBI | 3.9 (0) | 3.5 (4.0) |
| NIST-9 | 7.1 (0) | 6.7 (5.6) |

## 5.5 Summary

We have developed a novel filter-based representation technique for fingerprint verification. The technique exploits both the local and global characteristics in a fingerprint image to make a verification. Each fingerprint image is filtered in a number of directions and a fixed-length feature vector is extracted in the central region of the fingerprint. The feature vector (FingerCode) is compact and requires only 640 (or 896, depending on image size) bytes. The matching stage computes the Euclidean distance between the template FingerCode and the input FingerCode. With increasingly cheaper CPU cycles and use of special purpose DSP chips, the computation time in FingerCode extraction will become a nonissue. On MSU_DBI database of $2,672$ fingerprints from 167 different subjects, 4 impressions per finger, we are able

to achieve a verification accuracy better than a state-of-the-art minutiae-based fingerprint matcher in terms of Equal Error Rate (EER) and only marginally inferior at very low FARs (when the reject rate is not considered). A similar performance is observed on the more challenging NIST-9 database. This shows that the discriminatory power of the proposed representation is comparable to that of the minutiae-based representation. Note that the performance of neither the minutiae-based system nor the filterbank-based system is even close to the theoretical performance upper bound established in Chapter 2.

The filterbank approach suffers from a number of disadvantages and more research is needed in the following areas to improve the representation and matching: (*i*) The registration is based on the detection of the reference point. Even though our multi-resolution reference point location algorithm is accurate and handles the poor quality fingerprint images gracefully, it fails to detect the reference point in very low quality images leading to either a rejection of the image or even worse, a false rejection in the verification system. A filterbank approach that aligns the fingerprints based on the minutiae information can achieve a more reliable registration and will not reject any images due to the absence of the reference point. However, the representation thus extracted will not be translation and rotation invariant resulting in a longer matching time. (*ii*) The current implementation of the filterbank representation is not rotational invariant. The rotation is handled in the matching stage by rotating the FingerCode itself. However, due to quantization of the rotation space and generation of multiple alignment hypotheses, the false accepts increase. This problem can be addressed by estimating a frame of reference of the fingerprints. However, estimation

of a frame of reference in the fingerprints is a difficult problem because all fingerprints have circular ridges in the portion above the core point. (*iii*) Due to skin elasticity, there is non-linear distortion in the fingerprint images and even if the fingerprints are registered in location and orientation, all ridges in all sectors may not align. This problem can be partially addressed by estimating the local ridge frequency in each sector and normalizing each sector to a constant ridge frequency. (*iv*) the FingerCode representation does not have any explicit procedure to handle the noise in the fingerprint images due to the dryness/smudginess of the finger. Although the sectors are normalized to a constant mean and variance and then filtered using a bank of Gabor filters, large amount of noise changes the gray-level image characteristics and causes problems in the quantification of discriminatory information in sectors. The simple variance-based features proposed in this thesis perform well, have good discriminatory power, and degrade more gracefully than the minutiae-based features with noise in the fingerprint images. However, we believe that extraction of richer and more discriminatory features from the sectors in the filtered images should be explored to improve the matching performance. (*v*) The current implementation of filterbank representation extraction takes longer than a typical minutiae-extraction algorithm. Approximately 99% of the total compute time for verification ($\sim$ 3 seconds on a SUN ULTRA 10) for the images in the MSU_DBI database is taken by the convolution of the input image with 8 Gabor filters. The convolution operation can be made significantly faster by dedicated DSP processors or performing the filtering in the frequency domain. If the reference point is correctly located, the features are translation invariant and the rotation handled in the matching stage is very fast. As

a result, the matching process is extremely fast. ($vi$) The current matching algorithm is very simple, an implementation of a smarter matching algorithm should be able to improve the verification performance. For example, the match resulting from each sector can be weighed differently based on image quality and a quantitative measure of the nonlinear distortion in the sector. The verification system should also benefit from a matcher that can handle conflicting information in the fingerprints.

# Chapter 6

# Decision-level Fusion in Fingerprint Verification

The current fingerprint verification systems do not meet the low FAR requirements of several civilian applications due to the nonlinear deformation and noise present in fingerprint images. An efficient and effective method to improve the verification performance is to combine multiple fingerprint matchers, multiple templates, and multiple fingers. We propose a combination scheme that is optimal (in the Neyman-Pearson sense) when sufficient data are available to obtain reasonable estimates of the joint densities of classifier outputs. Four different fingerprint matching algorithms are combined using the proposed scheme to improve the accuracy of a fingerprint verification system. Experiments conducted on the MSU_DBI database confirm the effectiveness of the proposed integration scheme. At the same FAR, the FRR improves by $\sim 3\%$ at all operating points as compared to the best individual matcher. We further show that a combination of multiple impressions or multiple fingers improves

the FRR by more than 4% and 5%, respectively at the same FAR at all operating points. Analysis of the results provide some insight into the various decision-level classifier combination strategies.

## 6.1 Introduction

In some applications with a stringent performance requirement (e.g., very low FAR), no single biometric can meet the requirements due to the inexact nature of sensing, feature extraction, and matching processes. This has generated interest in designing multimodal biometric systems [107]. Multimodal biometric systems can be designed to operate in one of the following five scenarios (see Figure 6.1): (*i*) Multiple sensors: for example, optical, ultrasound, and capacitance based sensors are available to capture fingerprints. (*ii*) Multiple biometric system: multiple biometrics such as fingerprint and face may be combined [65, 90, 13]. (*iii*) Multiple units of the same biometric: one image each from both the irises, or both the hands, or ten fingerprints may be combined [17]. (*iv*) Multiple instances of the same biometric: for example, multiple impressions of the same finger [17], or multiple samples of the voice, or multiple images of the face may be combined. (*v*) Multiple representations and matching algorithms for the same input biometric signal: for example, combining different approaches to feature extraction and matching of fingerprints [20]. The first two scenarios require several sensors and are not cost effective. Scenario (*iii*) causes an inconvenience to the user in providing multiple cues and has a longer acquisition time. In scenario (*iv*), only a single input is acquired during verification and matched

with several stored templates acquired during the one-time enrollment process. Thus, it is slightly better than scenario $(iii)$. In our opinion, scenario $(v)$, combination of different representation and matching algorithm, is the most cost-effective way to improve biometric system performance.

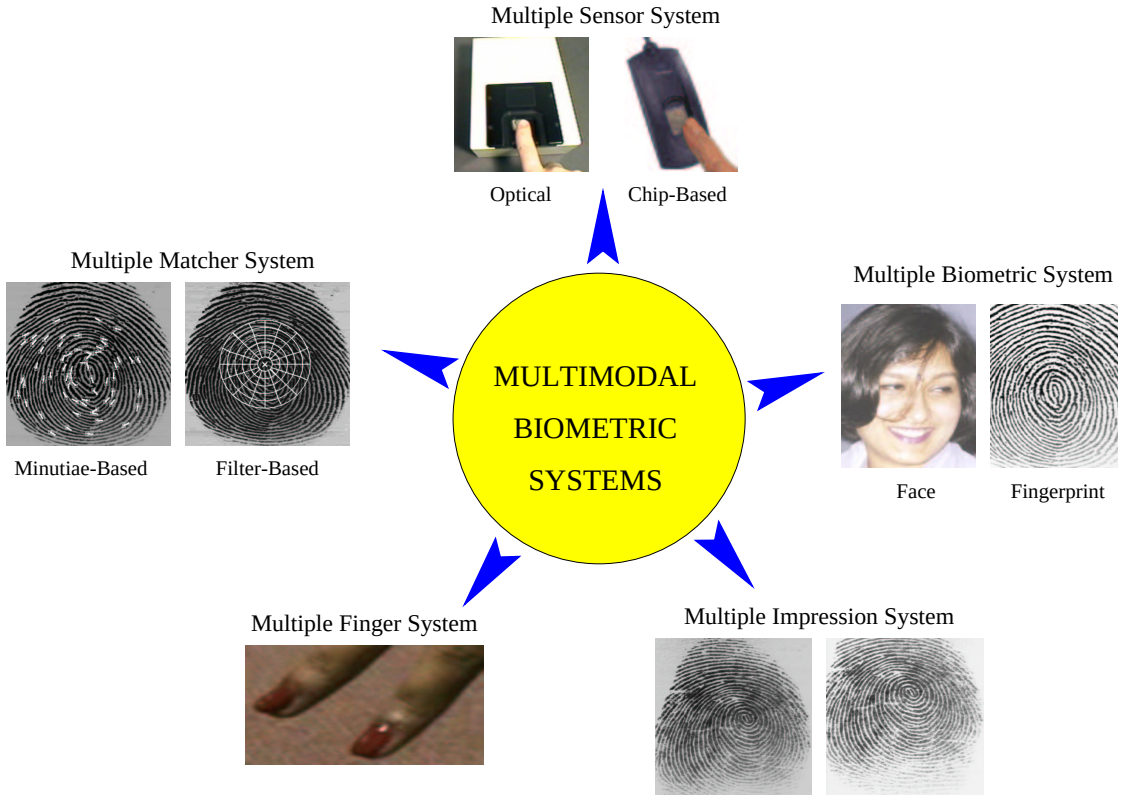



Figure 6.1: Various Multi-modal Biometric Systems [158].

We propose to use a combination of four different fingerprint-based biometric systems where each system uses different feature extraction and/or matching algorithms to generate a matching score which can be interpreted as the confidence level of the matcher. The proposed combination scheme operates at the decision-level. A combination at the feature level can result in a larger improvement. However, the feature extraction algorithms from different fingerprint verification system designers use proprietary code and typically only the confidence from the matcher is available.

A combination at the decision level is preferred over a combination at the abstract or the rank level because of more information being contained in the confidence value of a matcher. We combine the four different matching scores from four different matchers available to us to obtain the lowest possible FRR for a given FAR.

We also compare the performance of our integration strategy with the sum and the product rules [90]. Even though we propose and report results in scenarios (*iii*), (*iv*) and (*v*), our combination strategy could be used for scenarios (*i*) and (*ii*) as well.

## 6.2 Matcher Combination

A comprehensive list of classifier combination strategies can be found in [15, 90]. Jain et al. [15] summarize and categorize various classifier combination schemes based on architecture, selection and training of individual classifiers, and the characteristics of the combiner. A summary of various classifier combination schemes is shown in Table 6.1 [15]. However, a priori it is not known which combination strategy works better than the others and if so under what circumstances.

In this chapter we will restrict ourselves to a particular decision-level integration scenario where each classifier may select its own representation scheme and produces a confidence value as its output. A theoretical framework for combining classifiers in such a scenario has been developed by Kittler et al. [90]. The well known sum rule computes the sum of the aposteriori probabilities for each of the classes generated by individual classifiers generated by each matcher/classifier and makes the decision in favor of the class with the maximum sum. The product rule computes the prod-

Table 6.1: Confidence-level classifier combination schemes. A more detailed comparison can be found in [15].

| Combination Scheme | Trainable | Adaptable |
|---|---|---|
| Sum, mean, median | No | No |
| Product, min, max | No | No |
| Generalized ensemble | Yes | No |
| Adaptive weighting | Yes | Yes |
| Stacking | Yes | No |
| Logistic Regression | Yes | No |
| Dempster-Shafer | No | No |
| Mixture of local experts (MLE) | Yes | Yes |
| Hierarchical MLE | Yes | Yes |
| Bagging | Yes | No |
| Boosting | Yes | No |
| Neural tree | Yes | No |

uct of the aposteriori probabilities for each of the classes and makes the decision in favor of the class with the maximum product. The product rule implicitly assumes an independence of classifiers. The sum rule further assumes that the aposteriori probabilities computed by the respective classifiers do not deviate dramatically from the prior probabilities. The max rule, min rule, median rule, and majority vote rule have been shown to be special cases of the sum and the product rules [90]. Making these assumptions simplifies the combination rule but does not guarantee optimal results and hinders the combination performance. We follow Kittler et al.'s framework without making any assumptions about the independence of various classifiers.

## 6.3 Integration Strategy

Let us suppose that the test pattern $Z$ is to be assigned to one of the two possible classes, $w_0$ and $w_1$. Let us assume that we have $N$ classifiers, and the $i$th classifier outputs a single confidence value $\theta_i$ about class $w_1$ (the confidence for the class $w_0$ will be $1 - \theta_i$), $i = 1, 2, .., N$. Let us assume that the prior probabilities for the two classes are equal. The classifier combination task can now be posed as an independent (from the original $N$ classifier designs) classifier design problem with two classes and $N$ features ($\theta_i$, $i = 1, 2, .., N$).

### 6.3.1 Matcher Selection

It is a common practice in classifier combination to perform an extensive analysis of various combination strategies involving all the $N$ available classifiers. In feature selection, it is well known that the most informative $d$-element subset of $N$ conditionally independent features is not necessarily the union of the $d$ individually most informative features [85, 77, 166, 75]. Cover [167] argues that no non-exhaustive sequential $d$-element selection procedure is optimal, even for jointly normal features. He further showed that all possible probability of error orderings can occur among subsets of features subject to a monotonicity constraint. The statistical dependence among features causes further uncertainty in the $d$-element subset composed of the individually best features. One could argue that the combination strategy itself should pick out a subset of the classifiers that should be combined. However, we know in practice that the "curse of dimensionality" makes it difficult for a classifier to automatically

delete less discriminative features [8, 160]. Therefore, we propose a classifier selection scheme prior to classifier combination. We also propose to use the *class separation* statistic [82] as the feature effectiveness criterion. This statistic, $CS$, measures how well the two classes (imposter and genuine, in our case) are separated with respect to the feature vector, $X^d$, in a $d$-dimensional space, $R^d$.

$$CS(X^d) = \int_{R^d} |p(X^d|w_0) - p(X^d|w_1)|dx, \tag{6.1}$$

where $p(X^d|w_0)$ and $p(X^d|w_1)$ are the estimated distributions for the $w_0$ (imposter) and $w_1$ (genuine) classes, respectively. Note that $0 \leq CS \leq 2$.

We will use the class separation statistic to obtain the best subset of matchers using an exhaustive search of all possible $2^N - 1$ matcher subsets.

### 6.3.2 Non-parametric density estimation

Once we have selected the classifier subset containing $d$ ($d \leq N$, $N = 4$ in our case) classifiers, we develop our combination strategy. We do not make any assumptions about the form of the distributions for the two classes and use non-parametric methods to estimate the two (imposter and genuine) distributions. We will later show that this method is superior to a parametric approach which assumes a specific form of the density.

The Parzen window density estimate of a $d$-dimensional density function based

on $n$ i.i.d. observations (training samples) and a Gaussian kernel is given by [145]:

$$P(X) = \frac{1}{nh^d} \sum_{j=1}^{n} \left\{ \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma|^{\frac{1}{2}}} exp \left[ -\frac{1}{2h^2} (X - X_j)^t \Sigma^{-1} (X - X_j) \right] \right\}, \tag{6.2}$$

where $h$ is the window width. The covariance matrix, $\Sigma$, of the kernel is estimated from the $n$ training samples and $h \propto n^{\frac{-1}{d}}$. The value of $h$ is usually determined empirically. A large value of $h$ means a large degree of smoothing and a small value of $h$ means a small degree of smoothing of the estimated density. A rule of thumb states that for a small (large) number of training samples ($n$), window width should be large (small). Further, for a fixed $n$, the window width should be large (small) for large (small) number of features ($d$). When a large number of training samples are available, the density estimated using Parzen window approach is very close to the true density.

### 6.3.3 Decision Strategy

Bayes decision rule [145] is "optimal" for given prior probabilities and the class conditional densities. Bayes decision rule minimizes the total classification error (FAR + FRR) , that is, no other decision rule can yield a lower total error than the Bayes decision rule. However, in a fingerprint verification system, usually there is a constraint on the FAR dictated by the application of the verification system. In the case when the FAR is required to be below a prespecified value, the Neyman-Pearson decision rule is preferred which minimizes the FRR for a given FAR. The fingerprint verification is formulated as a hypothesis testing problem and the likelihood ratio

$L = P(X^d|w_0)/P(X^d|w_1)$ is used to construct the decision rule in our two-class problem: Decide $D_0$ (person is an imposter) for low values of $L$; decide $D_1$ (person is genuine) for high values of $L$. If $L$ is small, the input is more likely to come from class $w_1$; the likelihood ratio test rejects the null hypothesis for small values of $L$. The Neyman-Pearson lemma states that this test is optimal, that is, among all the tests with a given significance level, $\alpha$ (FAR), the likelihood ratio test has the maximum power (1-FRR). For a specified $\alpha$, $\lambda$ is the smallest constant such that $P\{L \leq \lambda\} \leq \alpha$. The type II error ($\beta$) is given by $P\{L > \lambda\}$. If we choose $\lambda = 1$, the Neyman-Pearson decision rule is equivalent to the Bayes decision rule under a $0 - 1$ loss function and equal priors. Since the designers of the verification system do not know in advance the particular application that the system will be used for, it is a common practice to report the performance of the system for a range of different FARs. We plot ROC curves using several different FARs and their corresponding FRR values obtained for a range of thresholds (values of $\lambda$).

## 6.4 Matching Algorithms

We have used four different fingerprint matching algorithms which can be broadly classified into two categories: ($i$) minutiae-based, and ($ii$) filter-based. The three minutiae-based and one filter-based algorithms are summarized in this section.

### 6.4.1 Hough Transform Based Matching (Algorithm *Hough*)

The fingerprint matching problem can be regarded as template matching [131]: given two sets of minutia features, compute their matching score. The two main steps of the algorithm are: (*i*) Compute the transformation parameters $\delta_x$, $\delta_y$, $\theta$, and $s$ between the two images, where $\delta_x$ and $\delta_y$ are translations along $x$- and $y$- directions, respectively, $\theta$ is the rotation angle, and $s$ is the scaling factor; (*ii*) Align two sets of minutia points with the estimated parameters and count the matched pairs within a bounding box; (*iii*) Repeat the previous two steps for the range of allowed transformations. The transformation that results in the highest matching score is believed to be the correct one. The final matching score is scaled between 0 and 99. Details of the algorithm can be found in [131].

### 6.4.2 String Distance Based Matching (Algorithm *String*)

Each set of extracted minutia features is first converted into polar coordinates with respect to an anchor point. The two-dimensional (2D) minutia features are, therefore, reduced to a one-dimensional (1D) string by concatenating points in an increasing order of radial angle in polar coordinates. The string matching algorithm is applied to compute the edit distance between the two strings. The edit distance can be easily normalized and converted into a matching score. This algorithm [11] can be summarized as follows: (*i*) Rotation and translation are estimated by matching ridge segment (represented as planar curve) associated with each minutia in the input image with the ridge segment associated with each minutia in the template image. The rotation

and translation parameters that result in the maximum number of matched minutiae pairs within a bounding box was used to define the estimated transformation and the corresponding minutiae are labeled as anchor minutiae, $\mathcal{A}_1$ and $\mathcal{A}_2$, respectively. (*ii*) Convert each set of minutia into a 1D string using polar coordinates anchored at $\mathcal{A}_1$ and $\mathcal{A}_2$, respectively; (*iii*) Compute the edit distance between the two 1D strings. The matched minutiae pairs are retrieved based on the minimal edit distance between the two strings; (*iv*) Output the normalized matching score (in the range of 0-99) which is the ratio of the number of matched-pairs and the number of minutiae points.

### 6.4.3 2D Dynamic Programming Based Matching (Algorithm *Dynamic*)

This matching algorithm is a generalization of the above mentioned string-based algorithm. The transformation of a 2D pattern into a 1D pattern usually results in a loss of information. Chen and Jain [152] have shown that fingerprint matching using 2D dynamic time warping can be done as efficiently as 1D string editing while avoiding the above mentioned problems with algorithm *String*. The 2D dynamic time warping algorithm can be characterized by the following steps: (*i*) Estimate the rotation between the two sets of minutia features as in Step 1 of algorithm *String*; (*ii*) Align the two minutia sets using the estimated parameters from Step 1; (*iv*) Compute the maximal matched minutia pairs of the two minutia sets using 2D dynamic programming technique. The intuitive interpretation of this step is to warp one set of minutia

to align with the other so that the number of matched minutiae is maximized; ($iv$) Output the normalized matching score (in the range of 0-99) which is based on only those minutiae that lie within the overlapping region. A penalty term is added to deal with unmatched minutia features.

### 6.4.4 Filterbank Based Matching (Algorithm *Filter*)

Chapter 5 describes our filterbank-based fingerprint verification algorithm. The distance score was inverted and normalized to a matching score between 0 and 99.

## 6.5 Experimental Results

One hundred images (about 4% of the database) were removed from the total of $2,672$ images in the MSU_DBI database because the filter-based fingerprint matching algorithm rejected these images due to failure in locating the center or due to a poor quality of the images. We matched all the remaining $2,572$ fingerprint images with each other to obtain $3,306,306$ ($2572 \times 2571/2$) matchings and called a matching genuine only if the pair contains different impressions of the same finger. Thus, we have a total of $3,298,834$ ($3,306,306 - 7,472$) imposter and $7,472$ genuine matchings per matcher from this database. For the multiple matcher combination, we randomly selected half the imposter matching scores and half the genuine matching scores for training (the Neyman-Pearson decision rule) and the remaining samples for test. This process was repeated ten times to obtain ten different training sets and ten corresponding independent test sets. All performances will be reported in terms of

*ROC* curves computed as an average of the ten *ROC* curves corresponding to the ten different training and test sets. For the multiple impression and multiple finger combinations, the same database of $3,298,834$ imposter and $7,472$ genuine matchings computed using the *Dynamic* matcher was used because it is the best individual matcher at low FARs.

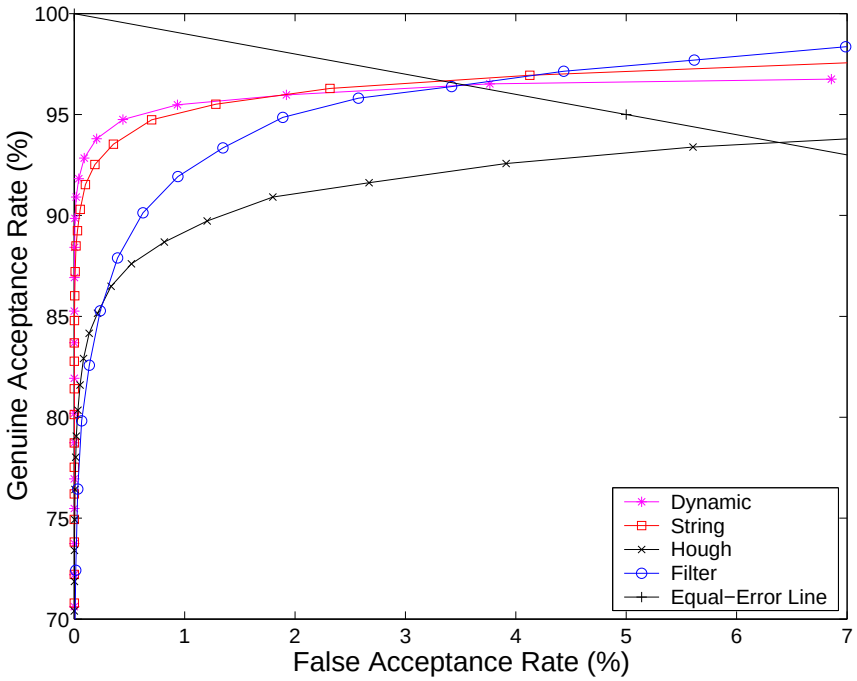
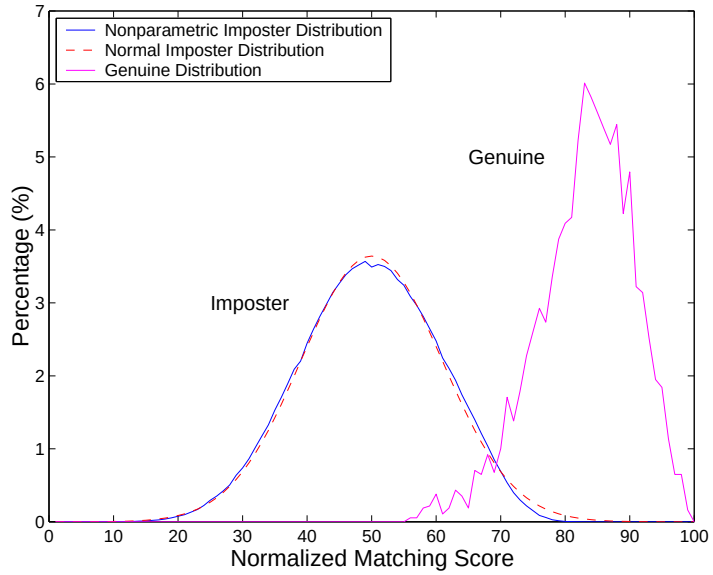
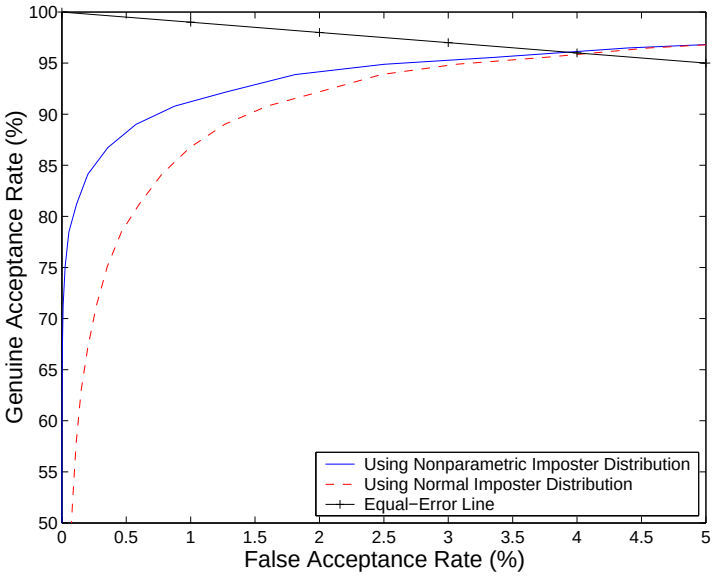



Figure 6.2: Performance of individual fingerprint matchers. The ROC curves have been averaged over ten runs.

The *ROC* curves computed from the test data for the four individual fingerprint matchers used in this study are shown in Figure 6.2. The class separation statistic computed from the training data was 1.88, 1.87, 1.85 and 1.76 for the algorithms *Dynamic*, *String*, *Filter*, and *Hough*, respectively, and is found to be highly correlated to the matching performance on the independent test set. Figure 6.2 shows that matcher *Filter* is better than the other three matchers at high FARs while it has the worst performance at very low FARs. Matcher *Hough* is the worst at most




(a)




(b)

Figure 6.3: Normal approximation to the imposter distribution for the matcher *Filter*. (a) Imposter and genuine distributions, (b) ROC curves. Visually, the Normal approximation seems to be good, but causes significant decrease in the performance compared to the nonparametric estimate of the imposter distribution at low FARs.

operating points except at very low FARs. At an equal error rate of about 3.5%, the matchers *Dynamic*, *String*, and *Filter* perform at the same level while the matcher *Hough* has an equal error rate of about 6.4%.

In general, biometrics applications demand very low error rates (e.g., FAR=0.01% and FRR=1.0%). Small errors in estimation of the imposter and genuine distributions can significantly effect the performance of a system. We will demonstrate this by approximating the imposter density with a normal density and using the empirical genuine density. This is because, visually, the imposter density looks like a normal density while the genuine density does not resemble a normal density. Consider the empirical genuine density and a normal approximation to the imposter density for the algorithm *Filter* shown in Figure 6.3(a). One would expect to get very accurate estimates of the parameters of a one-dimensional density from over 1.6 million data points. In fact, visually the normal approximation to the imposter density seems to fit the empirical density very well (see Figure 6.3(a)). As far as the equal error rate is concerned, using either the normal approximation or the nonparametric approximation of the imposter density give similar results. However, a significant decrease in performance is observed at low FARs when a normal approximation to the density is used in place of the nonparametric estimate (see Figure 6.3(b)). This is because the normal approximation to the imposter density has a heavier tail than the empirical density. To achieve the same low value of FAR, the system will operate at a higher threshold when the normal approximation to the density is used compared to when the nonparametric estimate of the density is used. The FRR, which is the area under the genuine density curve below the threshold, increases significantly. So, we would

like to stress that a parameterization of the density should be done with care.

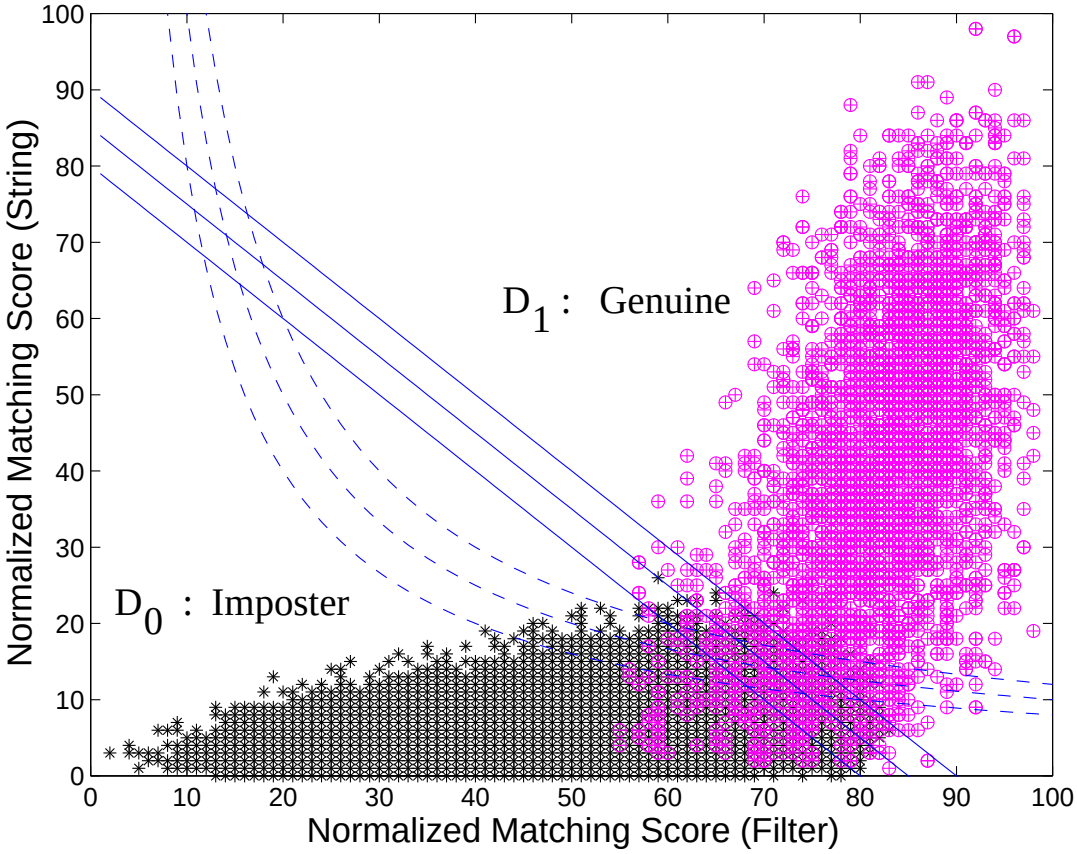



Figure 6.4: Plot of joint scores from matchers *String* and *Filter*. The solid lines denote the three sum rule decision boundaries corresponding to three different thresholds. The dotted lines denote the three product rule decision boundaries corresponding to three different thresholds.

Next, we combine the four available fingerprint matchers in pairs of two. It is well known in classifier combination studies that the independence of classifiers plays an important role in performance improvement [15]. A plot of the scores in a two-dimensional space from the training data for the *String* + *Filter* combination is shown in Figure 6.4. The correlation coefficient, $\rho$, between the matching scores can be used as a measure of diversity between a pair of matchers [110]. A positive value

Table 6.2: Combining two fingerprint matchers. $CS$ is the class separation statistic. $CS$ and $\rho$ are computed from the training data. Ranks by $EER$ (Equal Error Rate) are computed from the independent test data.

| Combination | $CS$ (rank) | rank by $EER$ | $\rho$ |
|---|---|---|---|
| $String + Filter$ | 1.95 (1) | 1 | 0.52 |
| $Dynamic + Filter$ | 1.95 (1) | 2 | 0.56 |
| $String + Dynamic$ | 1.94 (3) | 4 | 0.82 |
| $Hough + Dynamic$ | 1.93 (4) | 3 | 0.80 |
| $Hough + Filter$ | 1.91 (5) | 6 | 0.53 |
| $Hough + String$ | 1.90 (6) | 5 | 0.83 |

of $\rho$ is directly proportional to the measure of "dependence" between the scores from the two matchers. Table 6.2 lists the correlation coefficients for all possible pairings of the four available fingerprint matchers. It can be observed from this table that the minutiae-based fingerprint matchers have more dependence among themselves than with the filter-based fingerprint matcher. This is because the minutiae-based matchers are using the same features (minutiae set) and differ only in the matching algorithm.

To combine two fingerprint matchers, we first estimate the two-dimensional genuine and imposter densities from the training data. The two-dimensional genuine density was computed using the Parzen density estimation method. The value of window width ($h$) was empirically determined to obtain a smooth density estimate and was set at 0.01. We used the same value of $h$ for all the two-matcher combinations. As a comparison, the genuine density estimates obtained from the normalized histograms were extremely peaky due to unavailability of sufficient data (only about $3,780$ genuine matching scores were available in the training set to estimate a two-dimensional distribution in $10,000$ ($100 \times 100$) bins). However, for estimation of the

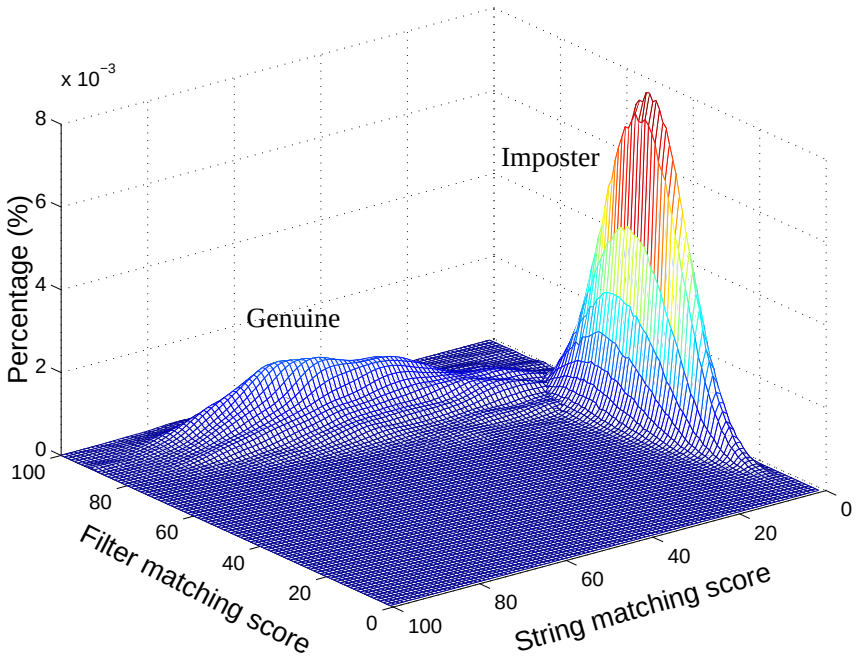



Figure 6.5: Two-dimensional density estimates for the genuine and imposter classes for $String + Filter$ combination. Genuine density was estimated using Parzen window ($h = 0.01$) estimator and the imposter density was estimated using normalized histograms.

two-dimensional imposter distribution, over 1.6 million matching scores were available. Hence, we estimated the two-dimensional imposter distribution by computing a normalized histogram using the following formula:

$$p(X^d|w_0) = \frac{1}{n}\sum_{j=1}^{n}\delta(X, X_j), \tag{6.3}$$

where $\delta$ is the delta function that equals 1 if the raw matching score vectors $X$ and $X_j$ are equal, 0 otherwise. Here $n$ is the number of imposter matchings from the training data. The computation time for Parzen window density estimate depends on $n$ and so, it is considerably larger than the normalized histogram method for large $n$. The smooth estimates of the two-dimensional genuine and imposter densities for

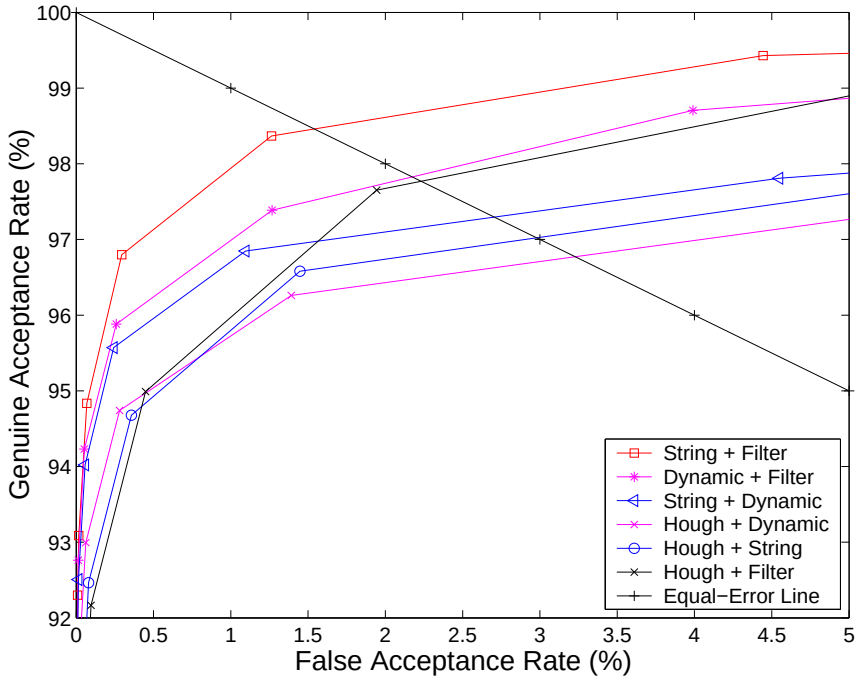



Figure 6.6: *ROC* curves for all possible two-matchers combinations.

*String* + *Filter* combination are shown in Figure 6.5. The class separation statistic for all pairs of matcher combination is shown in the second column of Table 6.2; the number in parenthesis is the predicted ranking of the combination performance based on *CS*. The actual ranking of performance obtained from the independent test set is listed in the third column marked *ROC* (see Figure 6.6 for *ROC* curves). As can be seen, the predicted ranking is very close to the actual rankings on independent test data.

The following observations can be made from the two-matcher combinations:

- Classifier combination improvement is directly related to the "independence" (lower values of $\rho$) of the classifiers.

- Combining two weak classifiers results in a large performance improvement.

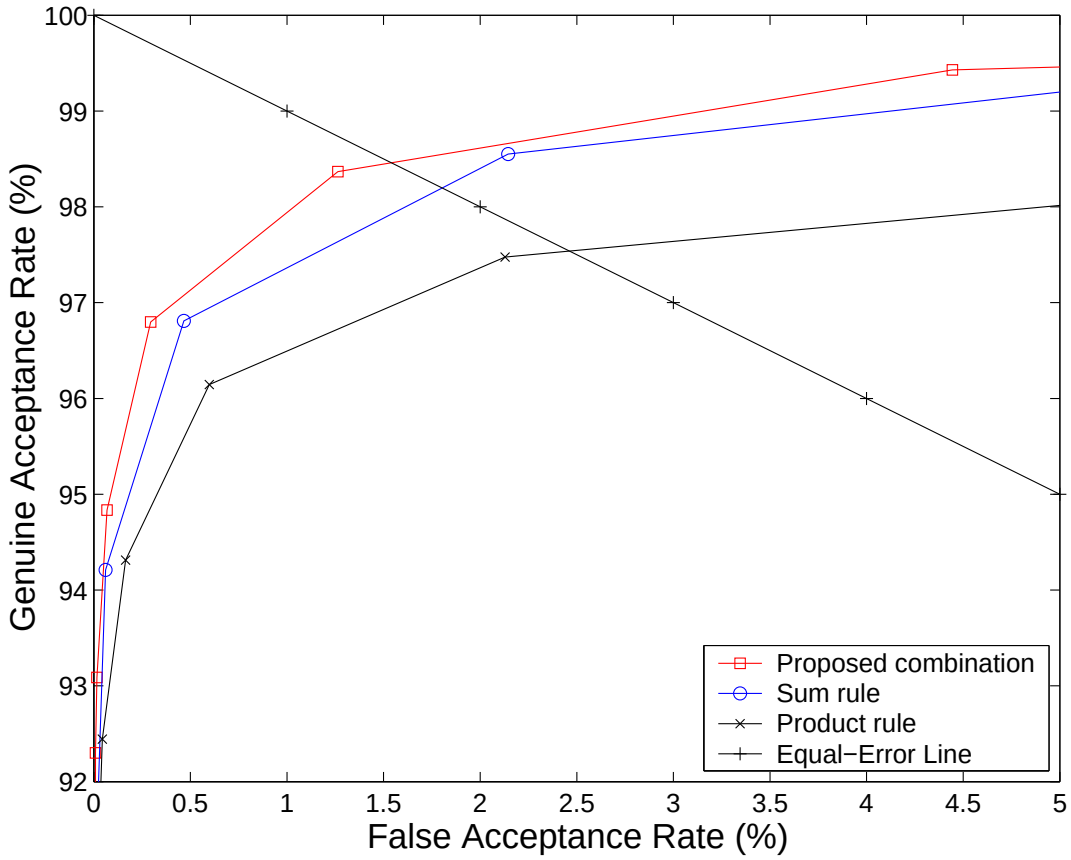



Figure 6.7: Comparison of the proposed combination scheme with the sum and the product rules for the $String + Filter$ combination.

- Combining two strong classifiers results in a small performance improvement.
- The two individually best classifiers do not form the best pair.

The proposed combination scheme either outperforms or maintains the performance of the sum rule and outperforms the product rule in all the two-, three-, and four-matcher combinations. However, we provide illustrations of the comparison in two-matcher combinations as it is easier to visualize the decision boundaries in two dimensions. We choose the $String + Filter$ combination which involves a strong and a weak classifier. The results of this combination and a comparison with the sum and the product rules is shown in Figure 6.7. By assuming that the errors in estimation

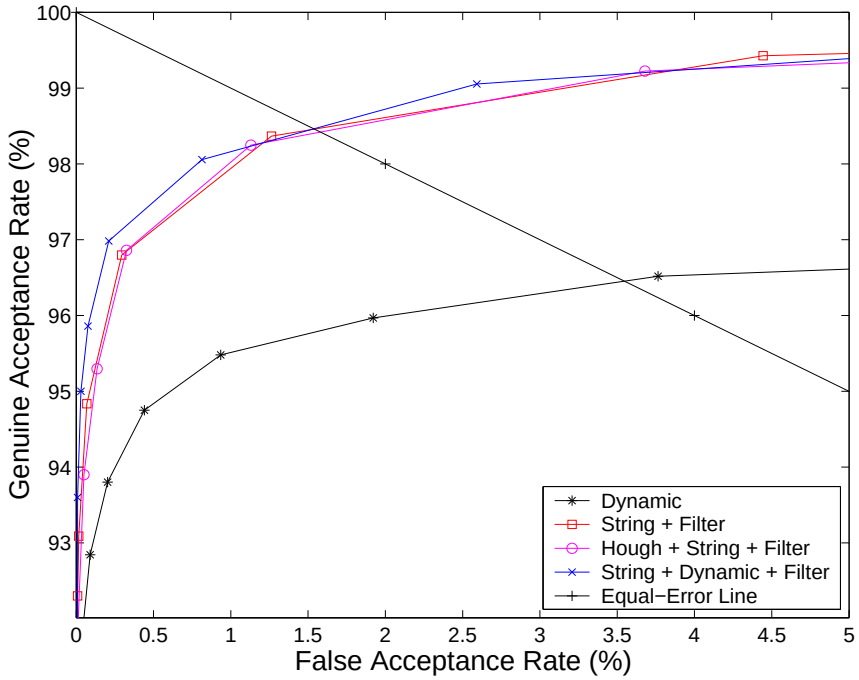



Figure 6.8: The performance of the best individual matcher *Dynamic* is compared with various combinations. The $String + Filter$ is the best two-matcher combination and $String + Dynamic + Filter$ is the best overall combination. Note that addition of the matcher *Hough* to the combination $String + Filter$ results in a degradation of the performance.

of aposteriori probabilities (matching scores) are very small, Kittler et al. [90] mathematically showed that the sum rule is less sensitive to these errors than the product rule. In our case, instead of considering the scores from two classifiers as estimates of aposteriori probability, we consider them as features in a separate classification problem. In such a case, the decision boundaries corresponding to the sum and the product rules can be drawn and visualized. In Figure 6.4 the decision boundaries corresponding to three different thresholds are shown for the sum and the product rules by solid and dotted lines, respectively. The product rule has a strong bias for low values of the two component classifier outputs. This is undesirable in most practical situations and the product rule is not expected to perform well in most cases. The

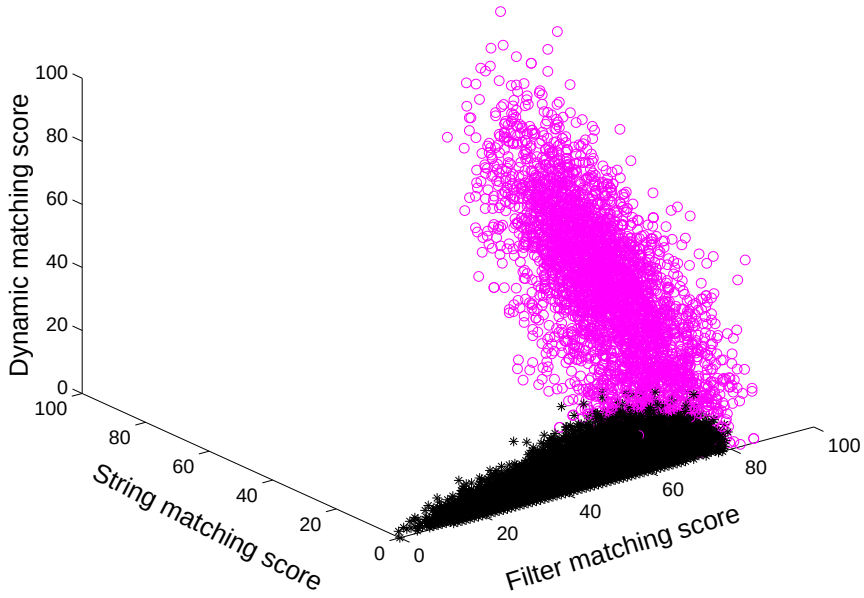



Figure 6.9: Matching scores for the best combination involving *String*, *Dynamic*, and *Filter* matchers. Visually, one can see a small overlap between the genuine ($\circ$) and the imposter ($*$) classes. The class separation statistic is 1.97 for the three-dimensional genuine and imposter densities estimated from these scores.

sum rule decision boundary is always a line with 135° slope and sum rule performs well only when combining two classifiers of equal strength (two weak or two strong classifiers). When a weak and a strong classifiers are combined, the decision boundary bends towards the axis of the strong classifier. A *weighted sum* rule weights the decisions from different classifiers differently for combination. Thus, the weighted sum rule can adapt the slope of its decision boundary but the decision boundary is still linear. The proposed technique can produce a decision boundary that is non-linear and is expected to perform better than the sum and the product rules. However, the disadvantage of the proposed technique is that it requires sufficient training data to obtain reasonable estimates of the densities while the sum rule is a fixed rule and does not require any training. The weighted sum rule can perform better than the sum

Table 6.3: Comparison of the performance of the best matcher combination with the best individual matcher. GAR refers to the genuine acceptance rate that is plotted on the ordinate of the *ROC* curves. We performed ten runs of the combination scheme with ten different splits of the database into training and test sets. The mean (*Mean*) and variance (*Var*) of the GAR values for three fixed values of FAR are reported.

| FAR | GAR *Dynamic* | GAR *String + Dynamic + Filter* | GAR Improvement |
|---|---|---|---|
| (%) | Mean (%) (Var (%)) | Mean (%) (Var (%)) | (%) |
| 1.00 | 95.53 (0.08) | 98.23 (0.02) | 2.70 |
| 0.10 | 92.96 (0.05) | 96.16 (0.04) | 3.20 |
| 0.01 | 90.25 (0.04) | 93.72 (0.05) | 3.47 |

rule but it is difficult to determine the weights. In summary, the proposed matcher combination scheme outperforms the commonly used sum rule and the product rule (Figure 6.7).

Finally, we combine the matchers in groups of three and then combine all the four matchers together. From the tests conducted on the independent data set, we make the following observations (see Figure 6.8).

- Adding a matcher may actually degrade the performance of classifier combination. This degradation in performance is a consequence of lack of independent information provided by the classifier being added and finite size of the training and test sets.

- Matcher selection based on a "goodness" statistic is a promising approach.

- Performance of combination of matchers is significantly better than the best individual matcher.

Among all the possible subsets of the four fingerprint matchers, the class separa-

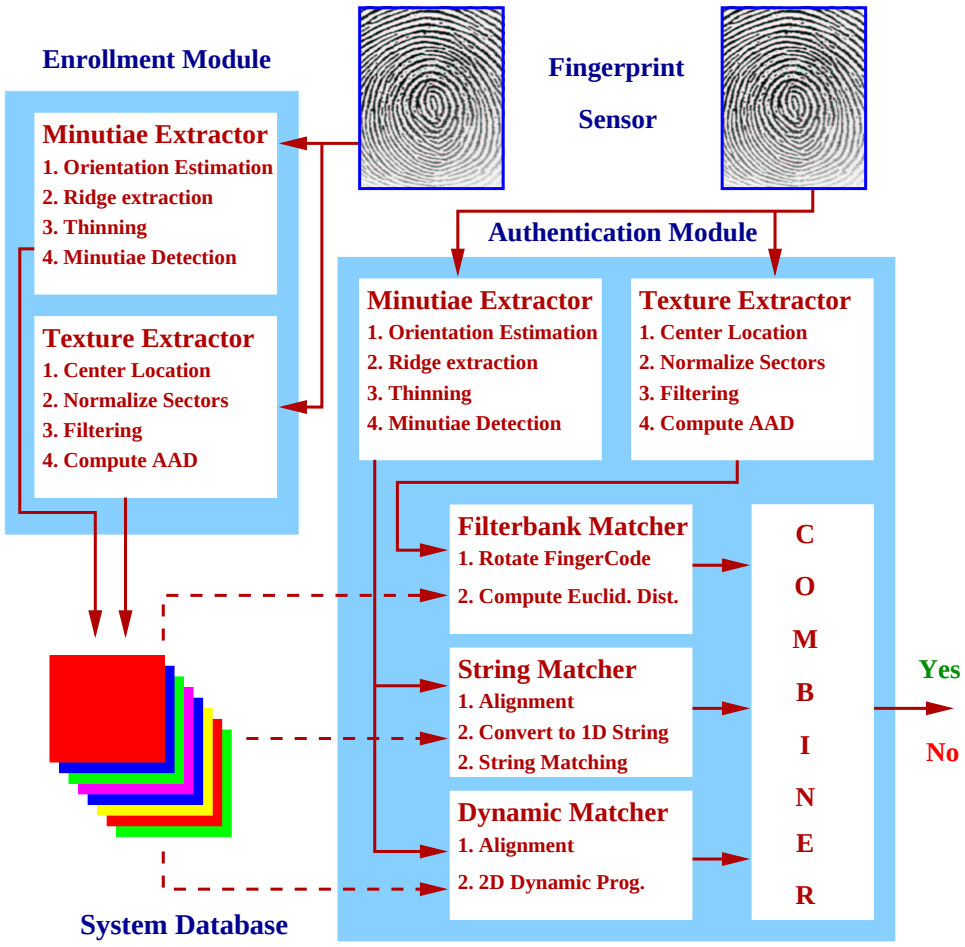



Figure 6.10: Proposed architecture of multi-modal biometrics system based on several fingerprint matchers.

tion statistic is the maximum for $String + Dynamic + Filter$ combination. Hence, our feature selection scheme selects this subset for the final combination and rejects the matcher $Hough$. This is consistent with the nature of the $Hough$ algorithm, which is basically the linear pairing step in algorithms $String$ and $Dynamic$, without the capability of dealing with elastic distortions. Therefore, $Hough$ does not provide "independent" information with respect to $String$ and $Dynamic$ matchers. Figure 6.9 shows the small overlap in the scores from the genuine and the imposter classes for the best combination involving fingerprint matchers $String$, $Dynamic$, and $Filter$.

(a) Finger 1, Impression 1

(b) Finger 1, Impression 2

(c) Finger 2, Impression 1

(d) Finger 2, Impression 2

Figure 6.11: Performance of matcher combination. (a) & (b) and (c) & (d) were misclassified by the three individual matchers *String*, *Dynamic*, and *Filter* as impostors, but correctly classified as genuine by the combination. Both the minutiae-based and filterbank-based matchers can not deal with large nonlinear deformations, however, a combination of matchers can overcome this.

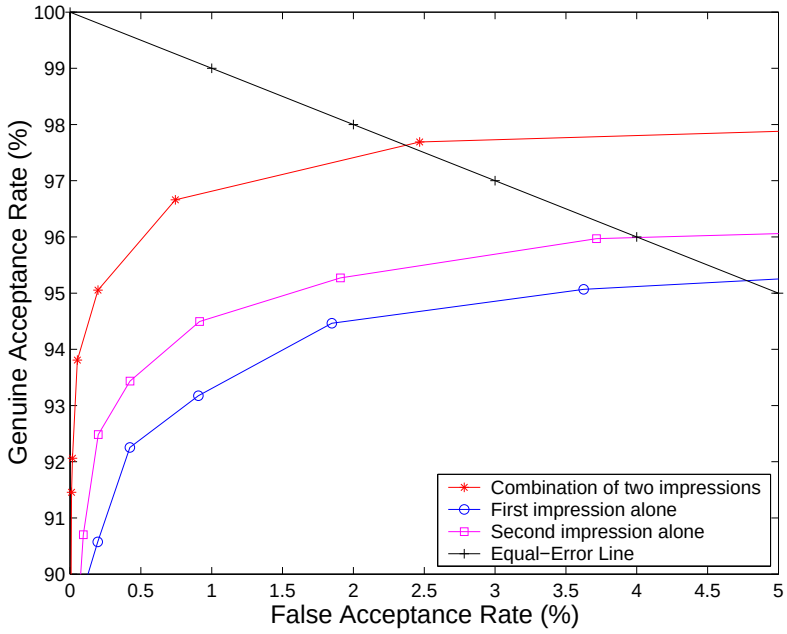
The performance of the various matcher combinations on an independent test set supports our claim that $String + Dynamic + Filter$ is the best combination. Figure 6.11 shows two pairs of images which were misclassified as impostors by all the three individual algorithms but correctly classified by the combined system.

Table 6.4: Equal error rate improvement due to combination of matchers.

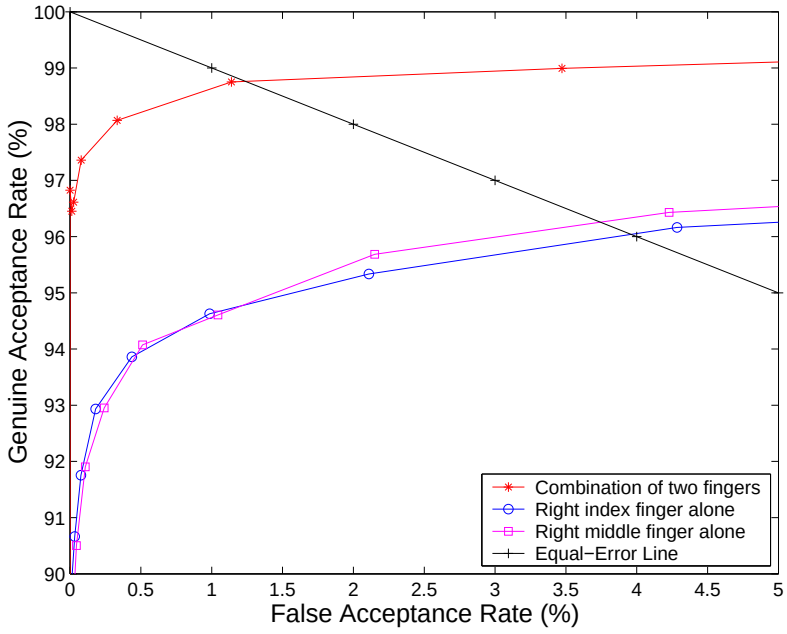
| | String | Dynamic | Filter | Combination |
|---|---|---|---|---|
| Equal Error Rate (%) | 3.9 | 3.5 | 3.5 | 1.4 |

The performance of the combined system is more than 3% better than the best individual matcher at low FARs (see Table 6.3). The equal error rate is more than 2% better than the best individual matcher (see Table 6.4). The matcher combination takes about 0.02 seconds on an Sun Ultra 10 in the test phase. This additional computational burden will have almost no effect on the overall matching time which will still be bounded by the slowest individual matcher ($Filter$) which takes about 3 seconds. Based on the experimental results presented above, we propose a multi-matcher biometric system design in Figure 6.10.

The performance improvement due to combination of two impressions of the same finger and the combination of two different fingers of the same person using the proposed strategy is shown in Figures 6.12(a) and (b), respectively. The best individual matcher *Dynamic* was used in these experiments. The correlation coefficient between the two scores from two different impressions of the same finger is 0.42 and between two different fingers of the same person is 0.68 and is directly related to the improvement in the performance of combination. The $CS$ for individual impressions




(a)




(b)

Figure 6.12: Performance improvement by using multiple impressions and multiple fingers. (a) Combining two impressions of the same finger, and (b) combining two fingers of the same person.

is 1.84 and 1.87, respectively, and for the combination $CS$ value is 1.95. The $CS$ for individual fingers is 1.87 and 1.86, respectively, and for the combination the $CS$ value is 1.98. Combination of two impressions of the same finger or two fingers of the same person using the proposed combination strategy is extremely fast. Therefore, the overall verification time is the same as the time taken by the matcher *Dynamic*.

## 6.6 Summary

We have presented a scheme for combining multiple matchers at decision-level in an optimal fashion. Our design emphasis is on matcher selection before arriving at the final combination. It was shown that one of the fingerprint matchers in the given pool of matchers is redundant and no performance improvement is achieved by utilizing this matcher in the combination. This matcher was identified and rejected by the matcher selection scheme. In case of a larger number of matchers and relatively small training data, a matcher may actually degrade the performance when combined with other matchers, and hence matcher selection is essential. We demonstrate that our combination scheme improves the false reject of a fingerprint verification system by more than 3% with no significant computational overhead. We also show that combining multiple impressions instances of a finger or multiple fingers is a viable way to improve the verification system performance. We observe that independence among various matchers is directly related to the improvement in performance of the combination.

# Chapter 7

# Fingerprint Feature Detection and Verification

Raw image data offer rich source of information for feature extraction and matching. For simplicity of pattern recognition system design, a sequential approach consisting of sensing, feature extraction and matching is conventionally adopted where each stage transforms a particular component of information relatively independently. The interaction between these modules is limited to one-way flow of control. Some of the errors in the end-to-end sequential processing can be easily eliminated especially for the feature extraction stage by revisiting the original image data. We propose a feedback path for the feature extraction stage, followed by a feature refinement stage for improving the matching performance. This performance improvement is illustrated in the context of a minutiae-based fingerprint verification system. We show that a minutia verification stage based on reexamining the gray-scale profile in a detected minutia's spatial neighborhood in the sensed image can improve the

matching performance by $\sim$ 2.2% equal error rate (point on the ROC where FAR is equal to FRR) on the GT database. Further, we show that a feature refinement stage which assigns a class label to each detected minutia (ridge ending and ridge bifurcation) before matching can also improve the matching performance by $\sim$ 1% equal error rate. A combination of feedback (minutia verification) in the feature extraction phase and feature refinement (minutia classification) improves the overall performance of the fingerprint verification system by $\sim$ 3%.

## 7.1 Introduction

(a) QI = 0.96 (b) QI = 0.53 (c) QI = 0.04

Figure 7.1: Sample images from the GT database with varying quality index (QI). 0 false minutiae were detected in (a), 7 in (b), and 27 in (c) by the automatic minutiae detection algorithm [11].

Most of the existing automatic fingerprint verification systems first detect the minutiae in a fingerprint image and then match the input minutiae set with the stored template [11, 56]. A typical algorithm described in [11] uses a sequential approach to

feature extraction. The feature extraction first binarizes the ridges in a fingerprint image using masks that are capable of adaptively accentuating the local maximum gray-level values along a direction normal to the local ridge direction. Minutiae are determined as points that have either one neighbor (ridge ending) or more than two neighbors (ridge bifurcation) in the skeletonized image (see Figure 1.2). However, the orientation estimation in a poor quality image is extremely unreliable, resulting in the detection of many false minutiae (see Figure 7.1). Several researchers have proposed minutia-pruning in the post-processing stage to delete spurious minutiae [11, 52, 136] but the pruning is based on rather ad-hoc techniques. In this chapter, we propose a feedback system for minutiae extraction which is based on an analysis of the gray scale profile in the neighborhood of potential minutiae. We also propose a feature refinement stage where the minutiae are classified into two major classes: ridge bifurcation and ending. The goal of the proposed feedback system (which we call minutia verification) is to learn the characteristics of minutiae in gray level images which can then be used to verify each detected minutia. This step can be used to replace the rather ad-hoc minutia-pruning stage used in [11]. Each detected minutia is filtered through this verification stage and is either accepted or rejected based on the learnt gray level characteristics in the neighborhood of a minutia. The minutia classifier is based on supervised Learning Vector Quantization (LVQ) [165]. We chose to use LVQ for our classification problem due to its fast learning speed and good performance. Also, public domain LVQ software can be downloaded from the web.

We show that the feature refinement (minutia classification into bifurcation and ending) can further improve the matching performance. We use a rule-based classifier

to classify a minutia into the two categories. The matching algorithm proposed in [11] is modified to match minutiae of the same type in the sensed image and the template. The modification of minutia matching algorithm used in [11] with minutia verification and minutia classification significantly improves the matching accuracy.

## 7.2 Minutia Verification

Our minutia verification algorithm can be divided into three stages; (*i*) feature extraction, (*ii*) training (learning the minutiae characteristics), and (*iii*) verification.

### 7.2.1 Feature Extraction

We use the minutiae detection algorithm developed by Jain et al. [11] for our study. Each detected minutia has the following three attributes: the $x$ and $y$ position and the direction of the ridge on which the minutia resides. We extract a $64 \times 64$ region centered at the $x$ and $y$ position of the minutia and oriented in the direction of the minutia. A minutia is captured in a $32 \times 32$ block in fingerprint images scanned at 500 *dpi*. A larger region of $64 \times 64$ was chosen to avoid the boundary problems in filtering. The extracted region is normalized to a constant mean and variance to remove the effects of sensor noise and gray-scale deformation because of finger pressure variations. In our experiments, we set the values of both the mean and variance to 100. We enhance the contrast of the ridges by filtering each $64 \times 64$ window with an appropriately tuned Gabor filter [19]. We set the frequency, $f$, of the Gabor filter to the average ridge frequency $(1/K)$, where $K$ is the average inter-ridge distance. The

average inter-ridge distance is approximately 10 pixels in a 500 *dpi* fingerprint image. The values of parameters $\delta_x$ and $\delta_y$ for Gabor filters were empirically determined and each is set to 4.0 (about half the average inter-ridge distance). Since the extracted region is in the direction of the minutia, the filter is tuned to 0° direction. We perform the filtering in the spatial domain with a mask size of $33 \times 33$. The Gabor response for each pixel in the region is scaled to eight gray levels. We extract a $32 \times 32$ region (see Figure 7.3) from the center of the $64 \times 64$ region to avoid boundary problems in normalization and filtering and concatenate the rows of the window to form a $1,024$-dimensional feature vector.

### 7.2.2 Training

In the training phase, minutiae and non-minutiae feature vectors are fed to a Learning Vector Quantizer to learn the characteristics of minutiae and non-minutiae regions. For the training phase, we need ground truth for the minutiae and non-minutiae points in a large number of fingerprints. So, we use the GT database that contains 900 fingerprint images from 269 different fingers and have the ground truth minutiae information provided by a fingerprint expert (see Figure 7.2). Other fingerprint databases that we have access to do not have the associated minutiae ground truth marked in them. The multiple impressions for each finger in the GT database were taken at different times. The images are of different sizes but all the images have been scanned at 500 *dpi* resolution with 256 gray levels. We use the first 450 images for training and the remaining 450 images from different fingers for testing.

(a) (b)

Figure 7.2: Examples of images in the GT database. The ground truth minutiae provided by an expert are marked on the image.

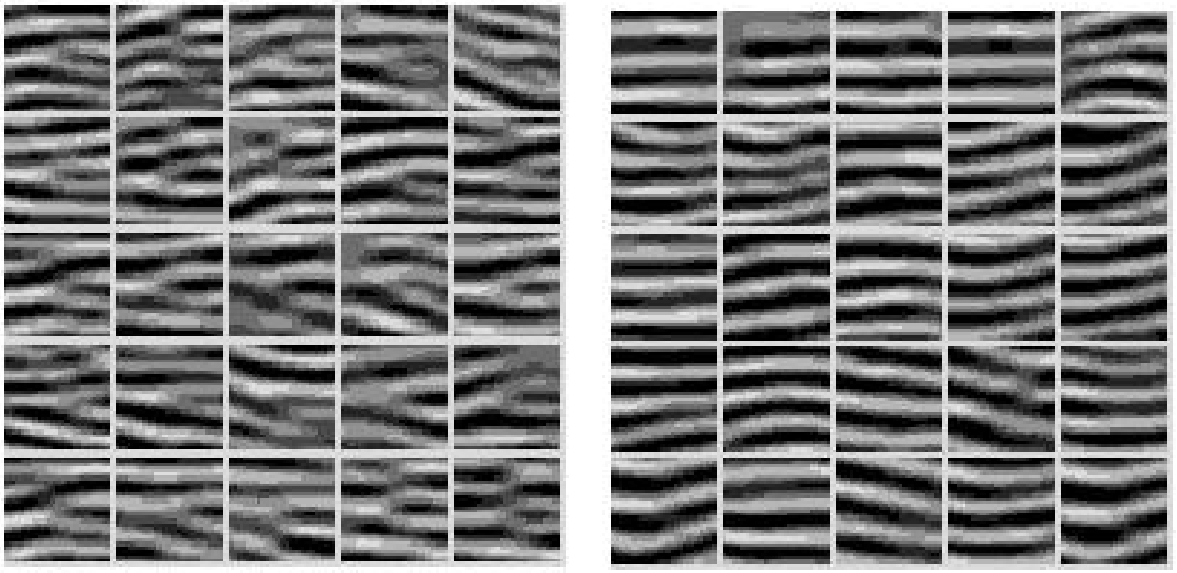
We extract approximately $15,000$ feature vectors (each feature vector has $1,024$ components) corresponding to all the true minutiae from the 450 images in the training database. We also extracted an equal number of negative samples (non-minutiae) by randomly sampling the images in the training set and making sure that there is no minutia in its immediate $32 \times 32$ neighborhood. For the true minutia, we use the direction of the minutia provided by the expert. For the negative examples, we compute the direction of the $32 \times 32$ block using the hierarchical orientation-field algorithm [11]. See Figure 7.3 for examples of minutiae and non-minutiae gray level profiles.

### 7.2.3 Testing

We use two methods to test the LVQ-based minutiae vs. non-minutiae classifier. In the first method, we evaluate the classifier using the ground truth minutia information in the test database. In the second method, we extract the minutiae from the test database using the minutiae extraction algorithm described in [11]. An automatically detected minutia may be slightly perturbed from its original location because of the noise introduced during the binarizing and thinning processes. So, we extract twenty five $32 \times 32$ windows in the neighborhood of each detected minutia and verify the presence of minutiae in each window. The decisions from the verification of these 25 windows are combined in a simple manner. If the classifier yields a positive verification for any of the 25 windows, the minutia is accepted. Figures 7.4 (a)-(c) compare the minutiae detection without pruning, with pruning, and with pruning replaced with minutia verification for a good quality fingerprint.

## 7.3 Minutia Classification

The American National Standards Institute proposes four classes of minutia: ending, bifurcation, trifurcation, and undetermined [23]. The most discriminable categories are ridge ending and bifurcation. A number of fingerprint matching algorithms do not even use minutia type information because of the difficultly in designing a robust classifier to identify minutiae type. However, we show that a consistent classification of minutia can indeed improve the overall matching performance. We use a rule-based minutiae classification scheme. In minutiae extraction algorithm, if a pixel in the



(a) (b)

Figure 7.3: Examples of gray level profiles in the neighborhood of (a) minutiae and (b) non-minutiae. These $32 \times 32$ subimages, scaled to 8 gray levels, are used for training an LVQ.

thinned image has only one neighbor then the minutia is classified as an ending, and if a pixel has more than 2 neighbors, then the minutia is classified as a bifurcation. The matching algorithm in [11] is modified to match minutiae endings only with minutiae endings and minutiae bifurcations only with minutiae bifurcations. In our experience, there are significantly more endings present in a typical fingerprint than bifurcations (according to a study conducted by Osterburg [94], the probability of occurrence of ridge endings is more than twice the probability of occurrence of ridge bifurcations). See Figure 7.4 (d) for minutia classification results.

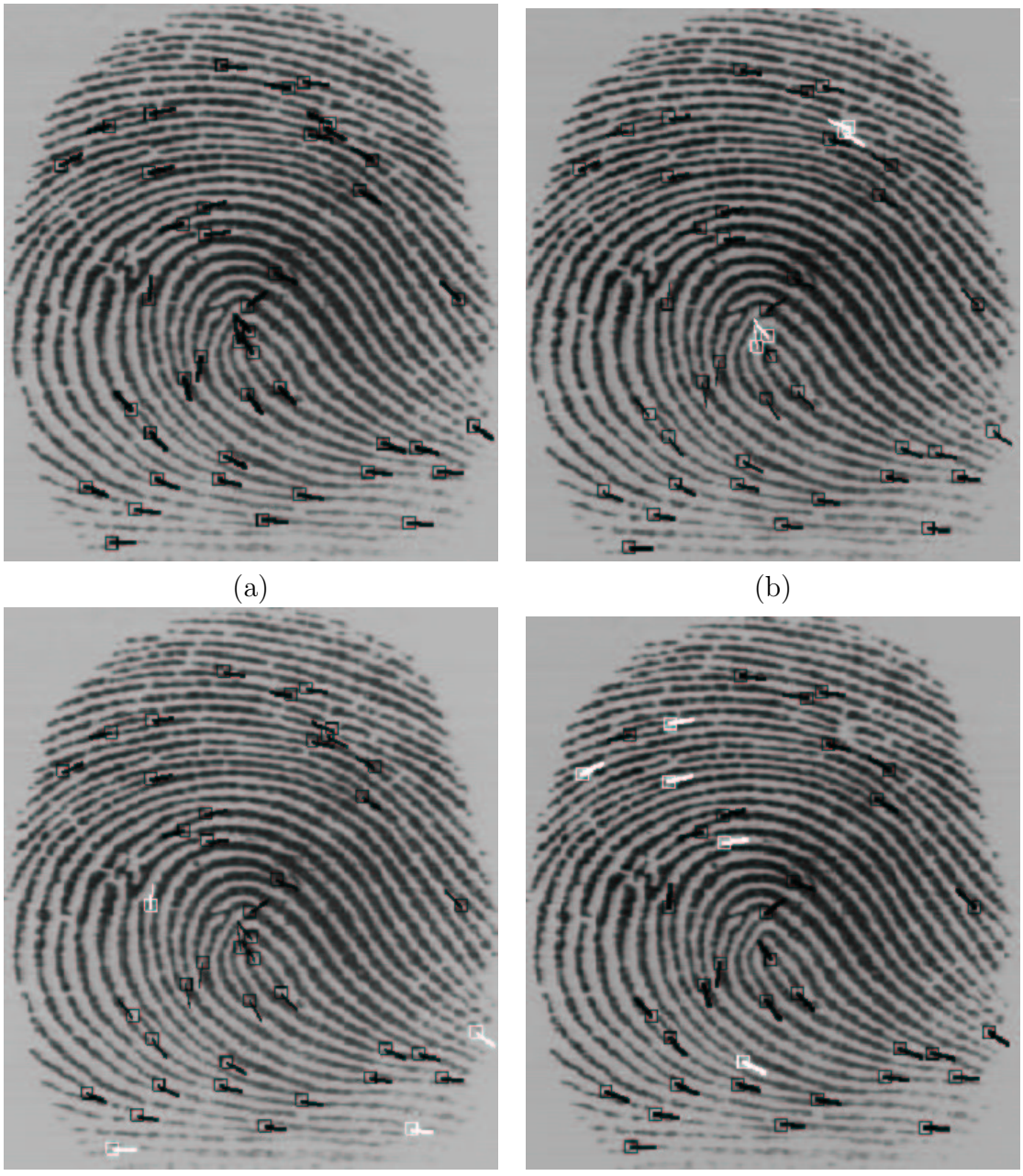



Figure 7.4: Minutiae detection and classification; (a) Minutiae detection using the algorithm in [11] without pruning, (b) results of minutia-pruning; minutiae marked in white were pruned, (c) result of minutia verification instead of pruning; minutiae marked in white were rejected, (d) result of classifying minutiae shown in (b); minutia bifurcations are marked in black and endings are marked in white.

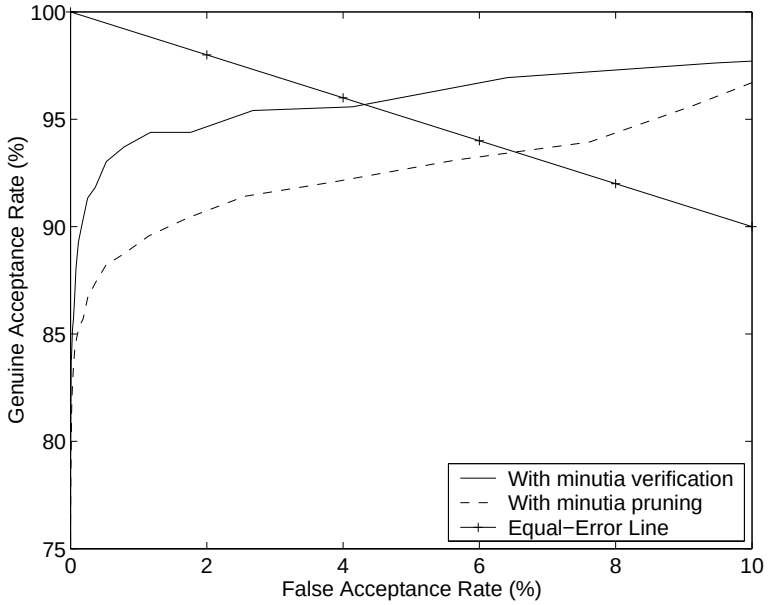



Figure 7.5: ROC for fingerprint matching when minutia verification is used.

## 7.4 Experimental Results

We first evaluated the performance of a minutiae-based fingerprint verification system [11] which incorporates the minutiae vs. non-minutiae classifier. Approximately $15,000$ $1,024$-dimensional feature vectors each for minutiae and non-minutiae were extracted from the the training database to design the LVQ classifier. The classifier was tested on an independent test set containing 450 images. The best performance of $\sim 95\%$ on the training data and $\sim 87\%$ on the test data was achieved with one hundred code book vectors per class. A real test for the utility of the minutiae verification module is the gain in matching accuracy when this module is incorporated in the matcher. So, we replaced the minutia-pruning stage in the algorithm in [11] with the proposed minutia verification stage. In the ROC curves shown in Figure 7.5, the dotted line represents the matching accuracy on the test set and the solid line represents the performance when the pruning stage in [11] is replaced with the proposed

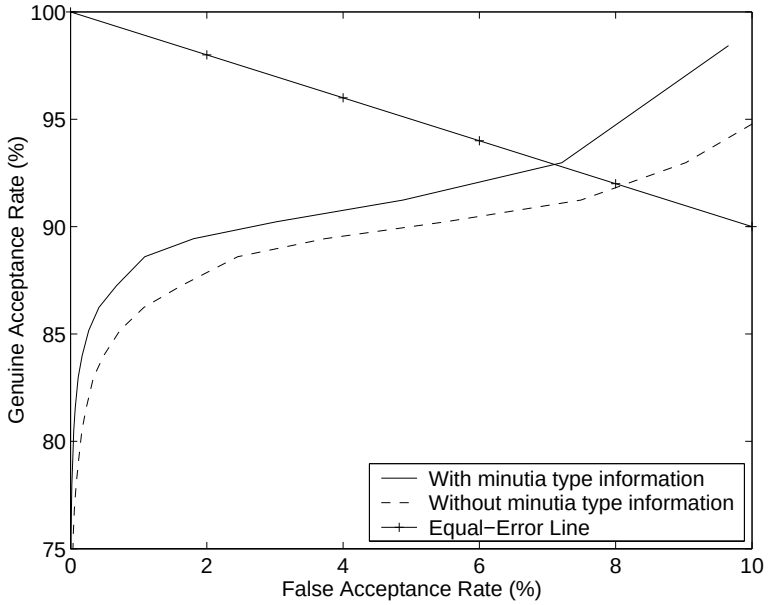



Figure 7.6: ROC for fingerprint matching when minutia classification is used.

minutia verification scheme. These ROC curves show that the overall performance of the fingerprint verification system increases by $\sim 3\%$ equal error rate.

The benefits of using minutia type information is illustrated in Figure 7.6. The solid line in the figure represents the performance when the minutia type information is used. Figure 7.7 shows the performance improvement when both minutia verification and minutiae classification are incorporated in the matcher. The classification is done before the verification but the classification information is not used during the verification. The performance of the fingerprint verification system in [11] is significantly improved by using the proposed minutia classification and minutiae verification modules.

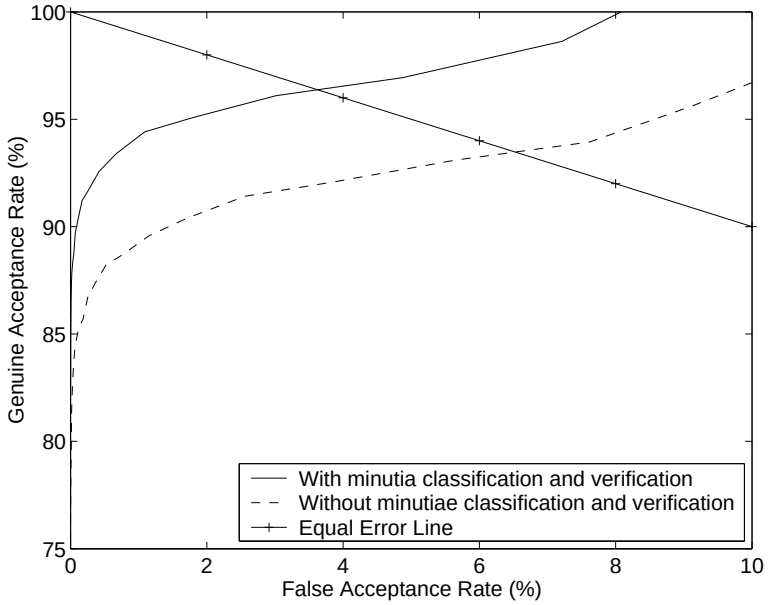



Figure 7.7: ROC for fingerprint verification when both minutia classification and verification are used.

## 7.5 Summary

We have shown that the performance of a minutiae-based fingerprint verification system can be improved by providing feedback in feature extraction (verification of each detected minutia by an analysis of grey-level profile in its spatial neighborhood in the original image). Performance can also be improved if the features are refined and more discriminable attributes (minutia type information) can be extracted and utilized in matching. The minutiae verification approach suffers from the problem of missed minutiae, i.e., the true minutiae in the fingerprint image that are missed by the feature extraction algorithm can not be recovered by the minutiae verification algorithm. Minutiae verification algorithm can only reject the falsely detected minutiae. Therefore, the minutiae detection algorithm should be operated at a very low false reject rate. We have accomplished this by removing the post-processing stage from

the feature extraction algorithm. However, there are still many missed minutiae in the fingerprint images that can not be recovered. The minutiae verification algorithm can also be applied on the whole or a subset of the image for minutiae detection. The minutiae detection algorithm will essentially need to examine a large number of candidates in the fingerprint image. With the current accuracy of the minutiae verification algorithm of $\sim 85\%$, a large number of errors will be made in the minutiae detection task. As a result, techniques to improve the current minutiae verification task should be explored further. In our training of the minutiae verification algorithm, the minutiae examples are representative of the total pattern variation in the minutiae types. However, the non-minutiae examples selected from random locations in the fingerprint images may not be representative of all the non-minutiae patterns. A more representative non-minutiae training set or a more clever method of using the training patterns for a more effective training should be explored to improve the performance of the minutiae verification algorithm.

# Chapter 8

# Conclusions and Future Work

## 8.1 Conclusions and Research Contributions

This thesis has concentrated on fingerprint-based biometric identification systems. Further, we have focused only on the core technology of fingerprint feature extraction, classification, and matching. There are a number of other very important issues in a fingerprint-based identification system including encryption, security of the fingerprint template, detection of fake fingers, and privacy concerns. These issues need to be addressed in a systematic way in developing a foolproof fingerprint-based identification system for a wide-scale deployment but are out of the scope of this thesis. The core fingerprint identification technology, i.e., fingerprint feature extraction, classification, and matching, are extremely important but challenging problems and even though several commercial systems exist for fingerprint verification, the performance (verification accuracy and time) needs to be improved for a wide adoption in authentication applications. One of the most fundamental questions one would like to ask

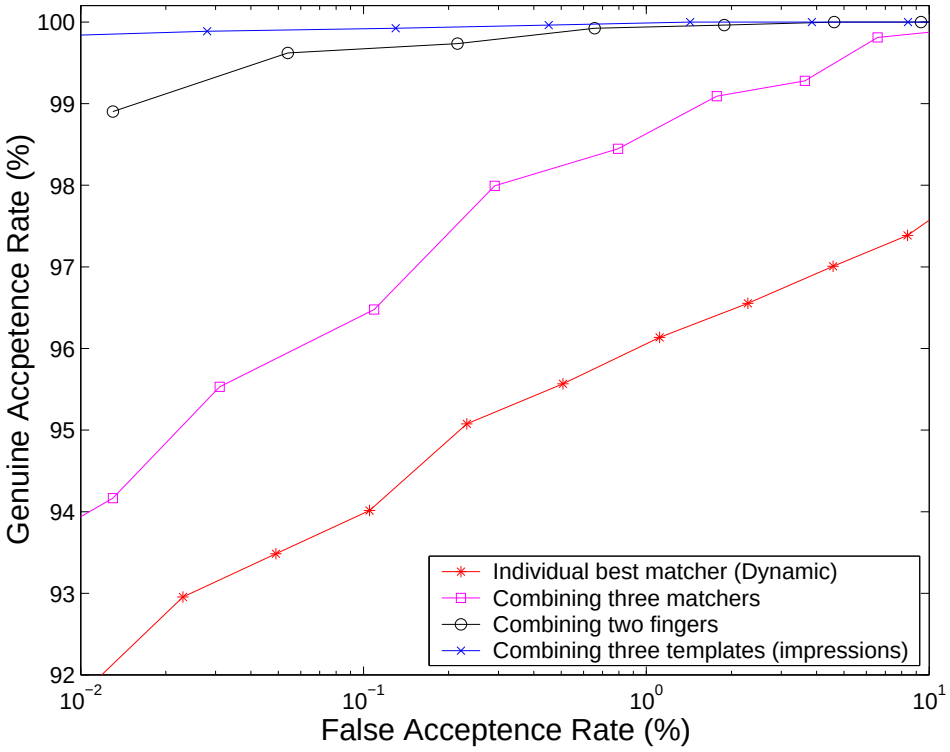



Figure 8.1: The best performance achieved on the MSU_DBI database. The minutiae extraction algorithm of Jain et al. [11] was modified by replacing its post processing stage with minutiae verification stage as described in Chapter 7. Three different matchers, namely, *String*, *Dynamic*, and *Filter*, two different fingers, and three different impressions for each finger of a person were combined. The genuine distribution was estimated using $2,640$ matchings and the imposter distribution was estimated using $95,920$ matchings. Note that the improvement in performance by combining multiple fingers is higher than combining multiple matchers or multiple templates (impressions). This is because different fingers provide the most "independent" information. A simple "sum rule" was used for the combination.

about a fingerprint authentication system is: what is the inherent discriminable information available in the fingerprints? Unfortunately, this question, if at all, has been answered in a very limited setting. In this thesis, we have quantitatively analyzed genetic and environmental factors influencing the information content in minutiae-based representation of fingerprints. This analysis established a performance limitation on automatic minutiae-based fingerprint identification due to the limited amount of information present in minutiae representation. Automatic fingerprint identification system designers, should therefore, explore non-minutiae-based fingerprint representations.

We have developed a novel filterbank-based representation for fingerprints. This representation is compact and has good discriminatory power. We have used this representation to achieve fingerprint classification and matching accuracies in line with the best accuracies reported in the literature. The primary advantage of our approach is its computationally attractive matching/indexing capability. For instance, if the translation and orientation normalized FingerCodes of all the enrolled fingerprints are stored as templates, the identification effectively involves a "bit" comparison. As a result, the identification time would be relatively insensitive to the database size. Further, our approach for feature extraction and matching is more amenable to hardware implementation than, say, a string-based minutiae matcher. We have proposed a general system design for decision-level matcher fusion that uses the optimal Neyman-Pearson decision rule and outperforms the combination strategies based on the assumption of independence among the matchers. We have proposed a multimodal biometric system design based on multiple fingerprint matchers. The use of the

proposed combination strategy in combining multiple matchers significantly improves the overall accuracy of the fingerprint-based verification system. The effectiveness of the proposed integration strategy is further demonstrated by building multi-modal biometric systems that combine two different impressions of the same finger or fingerprints of two different fingers. The proposed feature refinement and feedback stages in a minutiae-based feature extraction algorithm has been shown to improve the verification performance. The various techniques proposed in this thesis have significantly improved the overall performance of the fingerprint verification system (see Figure 8.1) and have contributed significantly in improving the state-of-the-art in fingerprint verification.

There are still a number of challenges in fingerprint verification. For example, almost all current fingerprint capture devices can be spoofed by some kind of a fake finger (e.g., tight fitting latex glove having an impression of somebody else's fingerprint). Fingerprint liveness detection is a difficult problem because the vital signs or liveliness identifiers often turn out to be more behavioral characteristics and tend to be volatile. However, the fingerprint capture devices and verification systems should strive to make it increasingly difficult to fake a finger by incorporating anti-spoofing measures into the hardware and software. The current combined (minutiae-based and filterbank-based) verification system still cannot deal with very poor quality fingerprints and large nonlinear distortion. An estimated 4% of the population including old people, asian women, and manual workers do not have good quality fingerprints and this poses a challenge to the matching system. Although, our fingerprint verification system has no difficulty in identifying children of any age, our tests were

conducted over a short period of time (three months). Children's fingers grow in size with age and the ridge characteristics such as the inter-ridge distance changes. If a child registers into the system today, the verification system will have difficulty in identifying him/her in a few years with the same template. This problem can be addressed by either a regular update of the child's template in the database or by incorporating the finger growth invariance into the matcher.

If we put all the advantages and disadvantages of fingerprint as a biometrics in perspective, we believe that the core technology of fingerprint verification (one-to-one matching) has achieved a performance (error rates and timing) that may be sufficient for several civilian applications and in the near future we should see fingerprints beings increasingly used in authentication systems as the cost of the fingerprint devices reduces further. The fingerprint classification (indexing) and identification (one-to-many matchings), on the other hand, have not reached sufficiently high accuracy for a wide-scale deployment. If an identification system has $N$ users in the database, then $N$ fingerprint matching are needed to be performed without any indexing. An efficient indexing technique should be able to reduce the number of matchings to $N'$ where $N' \leq N$. However, the foundation of an identification system lies on the core technology of fingerprint feature extraction and matching. Therefore, the feature extraction and matching algorithms need to be further improved in order to be used in identification systems. A number of future research directions to improve the filterbank-based as well as minutiae-based systems are given in the following section.

## 8.2 Future Directions

Our research can be expanded in the following areas:

- The registration in the FingerCode extraction is based on the detection of the reference point. Even though our multi-resolution reference point location algorithm is accurate and handles the poor quality fingerprint images gracefully, it fails to detect the reference point in very low quality images leading to either a rejection of the image or even worse, a false rejection in the verification system. A more robust feature extraction algorithm should not rely on a single reference point alone. As a possible solution, multiple reference point candidates can be located and representations corresponding all of these reference points can be stored as multiple templates. At the time of verification, match the input representation with each of the multiple representations and output the maximum matching score. As another possible solution, an alignment can be established using the minutiae features in a fingerprint. Such a system will not reject any images due to the absence of the reference point and perform well for the medium quality fingerprint images where the extracted minutiae can still be used to achieve an alignment. However, the representation thus extracted will not be translation and rotation invariant resulting in a longer matching time. To deal with very poor quality fingerprint images where an alignment based on the detected minutiae points can not be established, an alternate alignment technique based on some other features of fingerprints such as the orientation field should be explored.

- The current implementation of the filterbank representation is not rotation invariant. The rotation is handled in the matching stage by rotating the FingerCode itself. However, due to quantization of the rotation space and generation of multiple alignment hypothesis, the false accepts increase. This problem can be addressed by estimating a frame of reference of the fingerprints. However, estimation of a frame of reference in the fingerprints is a difficult problem because all fingerprints have circular ridges in the portion above the reference point.

- Due to skin elasticity, there is non-linear distortion in the fingerprint images and even if the fingerprints are registered in location and orientation, all ridges in all sectors may not align. This problem can be partially addressed by estimating the local ridge frequency in each sector and normalizing each sector to a constant ridge frequency. To further address the non-linear distortion problem, the tessellation can be distorted in a non-linear way according to the fingerprint distortion model proposed in [142].

- The FingerCode representation does not have any explicit procedure to handle the noise in the fingerprint images due to the dryness/smudginess of the finger. Although the sectors are normalized to a constant mean and variance and then filtered using a bank of Gabor filters, large amount of noise changes the gray-level image characteristics and causes problems in the quantification of discriminatory information in sectors. The simple variance-based features proposed in this thesis perform well, have good discriminatory power, and degrade more gracefully than the minutiae-based features with noise in the fingerprint

images. However, we believe that extraction of richer and more discriminatory features from the sectors in the filtered images should be explored to improve the matching performance.

- The current implementation of filterbank representation extraction takes longer than a typical minutiae-extraction algorithm. The convolution operation can be made significantly faster by dedicated DSP processors or performing the filtering in the frequency domain. These implementation issues need to be addressed to make the FingerCode matching system real-time.

- The current matching algorithm is very simple. An implementation of a smarter matching algorithm should be able to improve the verification performance. For example, the match resulting from each sector can be weighed differently based on image quality and a quantitative measure of the nonlinear distortion in the sector. The verification system should also benefit from a matcher that can handle conflicting information in the fingerprints.

- The current minutiae verification algorithm is applied on the minutiae extracted using the algorithm in [11] that detects the minutiae in the thinned binarized fingerprint ridges. The minutiae patterns that are learnt during the training can be used to detect the minutiae in the gray scale fingerprint image directly. However, the current implementation of the minutiae verification algorithm can not be used for the minutiae detection problem due to its poor accuracy. For example, consider a $320 \times 320$ pixels fingerprint image scanned at 500 *dpi* resolution. Our minutiae verification algorithm samples a $32 \times 32$ region around

each minutiae and cannot tolerate more than 8-pixel displacement in the minutiae location. Therefore, at least 400 ($4 \times \frac{320 \times 320}{32 \times 32}$) candidate minutiae locations in the fingerprint image will need to be sampled. With the current 87% accuracy of our minutiae verification algorithm, there will be 52 errors made by the minutiae identification algorithm in the image. In a typical $320 \times 320$ fingerprint image scanned at 500 *dpi* resolution containing $30 - 40$ minutiae on an average, 52 errors can result in missing all the correct minutiae on one extreme to a false detection of 52 minutiae on the other extreme. Therefore, techniques to improve the accuracy of the minutiae verification algorithm should be explored. At the same time, an intelligent scheme to apply the minutiae verification algorithm to only selected locations instead of the whole image should also be explored.

- The design of a core point learning and verification algorithm similar to the minutiae learning and verification algorithm described in this thesis should be explored to verify the detected reference point in the filterbank representation extraction algorithm. The current limitation in developing such an algorithm is the unavailability of large number of ground truth core examples.

- A number of people have speculated upon the nature of invariant information in the fingerprints. In particular, different researchers have granted a varying degree of latitude in the transformation invariance of the minutiae and based their matching algorithms on this hypotheses. For instance, some assume mostly rigid global transformation, others similarity transformation, while some others non-linear local transformations. However, there is no study supporting the ba-

sis for these hypotheses on which the entire matcher design relies. A study and quantization of minutiae transformation invariance information will be beneficial for the minutiae-based algorithms. Also, an estimate of the minutiae transformation invariance could also form a basis for the transformation invariance information for the fingerprints themselves. This study could be conducted using the $GT$ database which has 900 fingerprint images that have the minutiae location, orientation and correspondences between a pair of fingerprints marked by an expert.

# Bibliography

# Bibliography


[1] A. Alamansa and L. Cohen, "Fingerprint Image Matching by Minimization of a Thin-Plate Energy Using a Two-Step Iterative Algorithm with Auxiliary Variables," *Workshop on the Application of Computer Vision*, Palm Springs, California, December 4 - 6, 2000.

[2] A. C. Bovik, M. Clark, and W. S. Geisler, "Multichannel Texture Analysis Using Localized Spatial Filters," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 12, No. 1, pp. 55-73, January 1990.

[3] A. C. Bovik, N. Gopal, T. Emmoth, and A. Restrepo, "Localized Measurement of Emergent Image Frequencies by Gabor Wavelets," Special Issue on Wavelet Transforms and Multiresolution Signal Analysis, *IEEE Transactions on Information Theory*, Vol. IT-38, no. 3, pp. 691-712, March 1992.

[4] Access the Web with your face. http://www.miros.com/web_access_demo_page.htm.

[5] A. K. Hrechak and J. A. McHugh, "Automated Fingerprint Recognition Using Structural Matching," *Pattern Recognition*, Vol. 23, pp. 893-904, 1990.

[6] A. K. Jain, A. Ross, and S. Prabhakar, "Fingerprint Matching Using Minutiae and Texture Features", to appear in the *International Conference on Image Processing (ICIP)*, Greece, October 7-10, 2001.

[7] A. K. Jain, A. Ross, and S. Pankanti, "A Prototype Hand Geometry-Based Verification System", *2nd Int'l Conference on Audio- and Video-based Biometric Person Authentication*, Washington D.C., pp. 166-171, March 22-24, 1999.

[8] A. K. Jain and B. Chandrasekaran, "Dimensionality and Sample Size Considerations in Pattern Recognition Practice," in *Handbook of Statistics*, Vol. 2, P. R. Krishnaiah and L. N. Kanal (eds.), North-Holland, pp. 835-855, 1982,

[9] A. K. Jain and D. Zongker, "Feature Selection: Evaluation, Application, and Small Sample Performance", *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 19, No. 2, pp. 153-158, 1997.

[10] A. K. Jain and F. Farrokhnia, "Unsupervised Texture Segmentation Using Gabor Filters," *Pattern Recognition*, Vol. 24, No. 12, pp. 1167-1186, 1991.

[11] A. K. Jain, L. Hong, S. Pankanti, and Ruud Bolle, "An Identity Authentication System Using Fingerprints," *Proceedings of the IEEE*, Vol. 85, No. 9, pp. 1365-1388, 1997.

[12] A. K. Jain, L. Hong, and R. Bolle, "On-line Fingerprint Verification," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 19, No. 4, pp. 302-314, 1997.

[13] A. K. Jain, L. Hong, and Y. Kulkarni "A Multimodal Biometric System using Fingerprint, Face, and Speech", *Proc. 2nd Int'l Conference on Audio- and Video-based Biometric Person Authentication*, Washington D.C., pp. 182-187, 1999.

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

[15] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical Pattern Recognition: A Review", *IEEE Transactions on Patt. Anal. and Machine Intell.*, Vol. 22, No. 1, pp. 4-37, 2000.

[16] A. K. Jain and S. Pankanti, "Biometrics Systems: Anatomy of Performance", IEICE Trans. Fundamentals, Special issue on biometrics, Vol. E84-D, No. 7, July 2001.

[17] A. K. Jain, S. Prabhakar, and A. Ross, "Fingerprint Matching: Data Acquisition and Performance Evaluation", *MSU Technical Report TR99-14*, 1999.

[18] A. K. Jain, S. Prabhakar, and L. Hong, "A Multichannel Approach to Fingerprint Classification", *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 21, No. 4, pp. 348-359, 1999.

[19] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based Fingerprint Matching," *IEEE Trans. Image Processing*, Vol. 9, No. 5, pp. 846-859, May 2000.

[20] A. K. Jain, S. Prabhakar, and S. Chen, "Combining Multiple Matchers for a High Security Fingerprint Verification System", Pattern Recognition Letters, Vol 20, No. 11-13, pp. 1371-1379, November 1999.

[21] A. K. Jain, S. Prabhakar, and S. Pankanti, "Twin Test: On Discriminability of Fingerprints" *3rd International Conference on Audio- and Video-Based Person Authentication*, pp. 211-216, Sweden, June 6-8, 2001.

[22] A. Lumini, D. Maio and D. Maltoni, "Continuous vs Exclusive Classification for Fingerprint Retrieval", Pattern Recognition Letters, Vol. 18, No. 10, pp. 1027-1034, October 1997.

[23] American National Standard for Information Systems – Data Format for the Interchange of Fingerprint Information, Doc No. ANSINIST-CSL 1-1993, American National Standards Institute, New York, 1993.

[24] A. Newman, "Fingerprinting's Reliability Draws Growing Court Challenges," *The New York Times*, April 7, 2001.

[25] A. P. Fitz and R. J. Green, "Fingerprint Classification Using Hexagonal Fast Fourier Transform," *Pattern Recognition*, Vol. 29, No. 10, pp. 1587-1597, 1996.

[26] A. Ranade and A. Rosenfeld, "Point Pattern Matching by Relaxation," *Pattern Recognition*, Vol. 12, No. 2, pp. 269-275, 1993.

[27] A. Ross, A. K. Jain, and J. Z. Qian, "Information Fusion in Biometrics", *Proc. 3rd International Conference on Audio- and Video-Based Biometric Person Authentication*, pp. 354-359, Sweden, June 6-8, 2001.

[28] A. R. Rao, *A Taxonomy for Texture Description and Identification*, Springer-Verlag, New York, 1990.

[29] A. R. Roddy and J. D. Stosz, "Fingerprint Features-Statistical Analysis and System Performance Estimates", *Proc. IEEE*, Vol. 85, No. 9, pp. 1390-1421, 1997.

[30] A. Senior, "A Hidden Markov Model Fingerprint Classifier," *Proceedings of the 31st Asilomar conference on Signals, Systems and Computers*, pp. 306-310, 1997.

[31] A. Sibbald, "Method and Apparatus for Fingerprint Characterization and Recognition Using Auto-correlation Pattern," *US Patent 5633947*, 1997.

[32] A. Sherstinsky and R. Picard, "Restoration and Enhancement of Fingerprint Images Using M-lattice - A novel Non-linear Dynamical System," *Proc. 13th International Conf. on Pattern Recognition*, Jerusalem, Israel, Vol. 2, pp. 195-200, Oct. 1994.

[33] B. Bhanu, "A Triplet Based Approach for Indexing of Fingerprint Database for Identification", *Proc. 3rd International Conference on Audio- and Video-Based Biometric Person Authentication*, pp. 205-210, Sweden, June 6-8, 2001.

[34] B. G. Sherlock and D. M. Monro, "A Model for Interpreting Fingerprint Topology," *Pattern Recognition*, Vol. 26, No. 7, pp. 1047-1055, 1993.

[35] B. Moayer and K. Fu, "A Tree System Approach for Fingerprint Pattern Recognition," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 8 no. 3, pp. 376-388, 1986.

[36] B. Wentworth and H. H. Wilder, *Personal Identification*, R. G. Badger, Boston, 1918.

[37] C. Champod and P. A. Margot, "Computer Assisted Analysis of Minutiae Occurrences on Fingerprints", *Proc. International Symposium on Fingerprint Detection and Identification*, J. Almog and E. Spinger, editors, Israel National Police, Jerusalem, pp. 305, 1996.

[38] C. E. Thomos, "Method and Apparatus for Personal Identification", *US Patent 3704949*, 1972.

[39] C. Kingston, "Probabilistic Analysis of Partial Fingerprint Patterns", *Ph.D. Thesis*, University of California, Berkeley, 1964.

[40] C. F. Shu and R. C. Jain, "Vector Field Analysis For Oriented Patterns," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 16, No. 9, pp. 946-950, 1994.

[41] C. I. Watson and C. L. Wilson, "NIST Special Database 4, Fingerprint Database," National Institute of Standards and Technology, March 1992.

[42] C. I. Watson and C. L. Wilson, "NIST Special Database 9, Fingerprint Database," National Institute of Standards and Technology, March 1992.

[43] C. L. Wilson, G. T. Candela, and C.I. Watson, "Neural Network Fingerprint Classification," *J. Artificial Neural Networks*, Vol. 1, No. 2, pp. 203-228, 1993.

[44] C. L. Wilson, J. L. Blue, and O. M. Omidvar, "Training Dynamics and Neural Network Performance", *Neural Networks*, Vol. 10, No. 5, pp. 907-923, 1997.

[45] C. L. Wilson, J. L. Blue, and O. M. Omidwar, "Neurodynamics of Learning and Network Performance", *Journal of Electronic Imaging*, Vol. 6, No. 3, pp. 379-385, 1997.

[46] C. V. Kameshwar Rao and K. Black, "Type Classification of Fingerprints: A Syntactic Approach," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 2, No. 3, pp. 223-231, 1980.

[47] D. A. Stoney and J. I. Thornton, "A Critical Analysis of Quantitative Fingerprint Individuality Models", *Journal of Forensic Sciences*, Vol. 31, No. 4, Oct 1986, pp. 1187-1216.

[48] D. A. Stoney, "Distribution of Epidermal Ridge Minutiae," *American Journal of Physical Anthropology*, Vol. 77, pp. 367-376, 1988.

[49] D. A. Stoney, "A Quantitative Assessment of Fingerprint Individuality", University of California, Berkeley, *Ph.D. Thesis*, 1985.

[50] Daubert v. Merrell Dow Pharmaceuticals, 113 S. Ct. 2786 (1993).

[51] D. B. G. Sherlock, D. M. Monro, and K. Millard, "Fingerprint Enhancement by Directional Fourier Filtering," *Proc. Inst. Elect. Eng. Visual Image Signal Processing*, Vol. 141, No. 2, pp. 87-94, 1994.

[52] D. C. D. Hung, "Enhancement and Feature Purification of Fingerprint Images," *Pattern Recognition*, vol. 26, no. 11, pp. 1,661-1,671, 1993.

[53] D. Costello, "Families: The Perfect Deception: Identical Twins", *Wall Street Journal*, February 12, 1999.

[54] Digital Biometrics, Inc., Biometric Identification Products. Available at: http://www.digitalbiometrics.com/

[55] Digital Persona, Inc., Fingerprint-based Biometric Authentication. http://www.digitalpersona.com/

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

[57] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, "FVC2000: Fingerprint Verification Competition", *Proc. 15th International Conference Pattern Recognition*, Barcelona, September 3-8, 2000, http://bias.csr.unibo.it/fvc2000/.

[58] D. Maio, D. Maltoni, and S. Rizzi, "An Efficient Approach to On-Line Fingerprint Verification," *Proc. International Symposium on Artificial Intelligence*, pp. 132-138, Mexico, 1995.

[59] D. Marr, *Vision*, San Francisco, California, W. H. Freeman, 1982.

[60] D. Partrid and W. B. Yates, "Engineering Multiversion Neural-net Systems", *Neural Computation*, Vol. 8, pp. 869-893, 1996.

[61] E. C. Driscoll, C. O. Martin, K. Ruby, J. J. Russel, and J. G. Watson, "Method and Apparatus for Verifying Identity Using Image Correlation," *US Patent No. 5067162*, 1991.

[62] E. Newham, *The Biometric Report*. New York: SBJ Services, 1995. http://www.sjb.co.uk/.

[63] E. P. Richards, "Phenotype vs. Genotype: Why Identical Twins Have Different Fingerprints?", http://www.forensic-evidence.com/site/ID_Twins.html.

[64] E. R. Henry, Classification and Uses of Fingerprints, London: Routledge, pp. 54-58, 1900.

[65] E. S. Bigün, J. Bigün, B. Duc, and S. Fischer, "Expert Conciliation for Multimodal Person Authentication Systems by Bayesian Statistics", in *Proc. 1st Int'l Conf. on Audio Video-based Biometric Person Authentication*, pp. 291-300, Crans-Montana, Switzerland, March 1997.

[66] E. Splitz, R. Mountier, T. Reed, M. C. Busnel, C. Marchaland, P. L. Roubertoux, and M. Carlier, "Comparative Diagnoses of Twin Zygosity by SSLP Variant Analysis, Questionnaire, and Dermatoglyphics Analysis," *Behavior Genetics*, pp. 56-63, Vol. 26., No. 1, 1996.

[67] F. Alkoot and J. Kittler, "Experimental Evaluation of Expert Fusion Strategies", *Pattern Recognition Letters*, Vol. 20, No. 11-13, pp. 1361-1369, 1999.

[68] Federal Bureau of Investigation. *The Science of Fingerprints: Classification and Uses*, U.S. Government Printing Office, Washington D.C., 1984.

[69] Federal Bureau of Investigation. www.fbi.gov

[70] F. Galton, *Finger Prints*, London: McMillan, 1892.

[71] G. C. Stockman and A. K. Agrawala, "Equivalence of Hough Curve Detection to Template Matching," *Communications of the ACM*, Vol. 20, pp. 820-822, 1977.

[72] G. Giacinto, F. Roli, and G. Fumera, "Design of Effective Multiple Classifier Systems by Clustering of Classifiers", *Proc. 15th International Conference on Pattern Recognition (ICPR)*, Barcelona, September 3-8, Vol. 2, pp. 160-163, 2000.

[73] G. J. Tomko, "Method and Apparatus for Fingerprint Verification", *US Patent 4876725*, 1989.

[74] G. L. Marcialis, F. Roli, and P. Frasconi, "Fingerprint Classification by Combination of Flat and Structural Approaches", *Proc. 3rd International Conference on Audio- and Video-Based Biometric Person Authentication*, pp. 241-246, Sweden, June 6-8, 2001.

[75] G. S. Fang, "A Note on Optimal Selection of Independent Observables", *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. SMC-9, No. 5, pp. 309-311, 1979.

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

[77] G. T. Toussaint, "Note on Optimal Selection of Independent Binary-valued Features for Pattern Recognition", *IEEE Trans. Inform. Theory*, Vol. IT-17, p. 618, 1971.

[78] H. C. Lee and R. E. Gaensslen (editors), *Advances in Fingerprint Technology*, Elsevier, New York, 1991.

[79] H. Cummins and Charles Midlo, *Fingerprints, Palms and Soles: An Introduction to Dermatoglyphics*. Dover Publications, Inc., New York, 1961.

[80] H. Cummins, W. J. Waits, and J. T. McQuitty, "The Breadths of Epidermal Ridges on the Fingertips and Palms: A Study of Variations," *American Journal of Anatomy*, Vol. 68, pp. 127-150, 1941.

[81] Identix Incorporated. www.identix.com

[82] I.-S. Oh, J.-S Lee, and C. Y. Suen, "Analysis of Class Separation and Combination of Class-Dependent Features for Handwriting Recognition", *IEEE Trans. Patt. Anal. and Machine Intell.*, Vol. 21, No. 10, pp. 1089-1094, 1999.

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

[84] J. Bigun, G. H. Granlund, and J. Wiklund, "Multidimensional Orientation Estimation with Applications to Texture Analysis and Optical Flow," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 13, No. 8, pp. 775-790, 1991.

[85] J. D. Elashoff, R. M. Elashoff, and G. E. Goldman, "On the Choice of Variables in Classification Problems with Dichotomous Variables", *Biometrika*, Vol. 54, pp. 668-670, 1967.

[86] J. G. Daugman, "High Confidence Recognition of Persons by a Test of Statistical Independence," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 15, No. 11, pp. 1148-1161, 1993.

[87] J. G. Daugman, "Two-Dimensional Spectral Analysis of Cortical Receptive Field Profiles," *Vision Res.*, Vol. 20, pp. 847-856, 1980.

[88] J. G. Daugman, "Uncertainty Relation for Resolution in Space, Spatial Frequency, and Orientation Optimized by Two-Dimensional Visual Cortical Filters," *J. Opt. Soc. Amer. A*, Vol. 2, pp. 1160-1169, 1985.

[89] J. G. Daugman and G. O. Williams, "A Proposed Standard for Biometric Decidability," in *Proc. CardTech/SecureTech Conf.*, pp. 223-234, Atlanta, GA, 1996.

[90] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On Combining Classifiers", *IEEE Trans. on Patt. Anal. and Machine Intell.*, Vol. 20, No. 3, pp. 226-239, 1998.

[91] J. L. Wayman, "Daubert Hearing on Fingerprinting: When Bad Science Leads to Good Law: The Disturbing Irony of the Daubert Hearing in the Case of U.S. V. Byron C. Mitchell", http://www.engr.sjsu.edu/ biometrics/publications_daubert.html

[92] J. L. Wayman, "Multi-finger Penetration Rate and ROC Variability for Automatic Fingerprint Identification Systems", *Technical Report*, San Jose State University, 1999.

[93] J. L. Wayman, "Technical Testing and Evaluation of Biometric Identification Devices," In *Biometrics: Personal Identification in Networked Society*, Anil K. Jain, Ruud Bolle, and S. Pankanti (editors), Kluwer Academic publishers, pp. 345-368, 1999.

[94] J. Osterburg, T. Parthasarathy, T. E. S. Raghavan, and S. L. Sclove, "Development of a Mathematical Formula for the Calculation of Fingerprint Probabilities Based on Individual Characteristics", Journal of the American Statistical Association, Vol 72, No. 360, pp. 772-778, 1977.

[95] J. P. Riganati and V. A. Vitols, "Automatic Pattern Processing System", *US Patent 4151512*, 1979.

[96] J. Ton and A. K. Jain, "Registering Landsat Images by Point Matching," *IEEE Trans. Geosci. Remote Sensing*, Vol. 27, No. 5, pp. 642-651, 1989.

[97] K. Karu and A. K. Jain, "Fingerprint Classification," *Pattern Recognition*, Vol. 29, No. 3, pp. 389-404, 1996.

[98] K. Pearson, "Galton's Work on Evidential Value of Fingerprints", *Sankhya: Indian Journal of Statistics*, Vol. 1, No. 50, 1933.

[99] K. Woods, W. P. Kegelmeyer Jr., and K. Bowyer, "Combination of Multiple Classifiers Using Local Accuracy Estimates", *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 19, No. 4, pp. 405-410, 1997.

[100] L. Amy, "Valeur de la Preuve en Dactyloscopie-I" Journal de la Societe de Statistique de Paris 87, pp. 80-87, 1946.

[101] L. Amy, "Valeur de la Preuve en Dactyloscopie-I" Journal de la Societe de Statistique de Paris 88, pp. 189-195, 1947.

[102] L. Amy, "Recherches Sur L'identification des Traces Papillaries", Annales de Medecine Legale, Vol. 28, No. 2, pp. 96-101, 1948.

[103] L. Coetzee and E. C. Botha, "Fingerprint Recognition in Low Quality Images," *Pattern Recognition*, Vol. 26, No. 10, pp. 1141-1460, 1993.

[104] L. Hong, "Automatic Personal Identification Using Fingerprints", Ph. D. Thesis, Department of Computer Science and Engineering, Michigan State University, East Lansing, 1998.

[105] L. Hong and A. K. Jain, "Classification of Fingerprint Images," *11th Scandinavian Conference on Image Analysis*, June 7-11, Kangerlussuaq, Greenland, 1999.

[106] L. Hong and A. K. Jain, "Integrating Faces and Fingerprints For Personal Identification," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol.20, No.12, pp 1295-1307, 1998.

[107] L. Hong, A. K. Jain and S. Pankanti, "Can Multibiometrics Improve Performance?", Proceedings AutoID'99, Summit, NJ, pp. 59-64, Oct 1999.

[108] L. Hong, Y. Wan, and A. K. Jain, "Fingerprint Image Enhancement: Algorithm and Performance Evaluation," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 20, No. 8, pp. 777-789, 1998.

[109] L. I. Kuncheva, "A Theoretical Study on Expert Fusion Strategies", *IEEE Transactions on Patt. Anal. Machine Intell.*, submitted 2000.

[110] L. I. Kuncheva, C. J. Whitaker, "Measures of Diversity in Classifier Ensembles", *submitted to Machine Learning*, 2000.

[111] L. Lam and C. Y. Suen, "Optimal Combination of Pattern Classifiers", *Pattern Recognition Letters*, Vol. 16, pp. 945-954, 1995.

[112] L. O'Gorman, "Fingerprint Verification," in *Biometrics: Personal Identification in a Networked Society*, A. K. Jain, R. Bolle, and S. Pankanti (editors), Kluwer Academic Publishers, pp. 43-64, 1999.

[113] L. O'Gorman and J. V. Nickerson, "An Approach to Fingerprint Filter Design", *Pattern Recognition*, Vol. 22, No. 1, 29-38, 1989.

[114] L. Xu, A. Krzyzak, and C. Y. Suen, "Methods for Combining Multiple Classifiers and Their Applications to Handwriting Recognition", *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 22, No. 3, pp. 418-435, 1992.

[115] M. Adhiwiyogo, S. Chong, J. Huang, and W. Teo, "Fingerprint Recognition", Final Report 18-551 (Spring 1999), http://www.ece.cmu.edu/ ee551/ Old_projects/projects/s99_19/ finalreport.html

[116] M. Clark, A. C. Bovik, and W. S. Geisler, "Texture Segmentation Using Gabor modulation/ demodulation," *Pattern Recognition Letters*, Vol. 6, pp. 261-267, September 1987.

[117] M. D. Eibert, "Human Cloning: Myths, Medical Benefits and Constitutional Rights", U&I Magazine, Winter 1999. Available at http://www.humancloning.org/users/infertil/humancloning.htm.

[118] M. Eshera and K. S. Fu, "A Similarity Measure Between Attributed Relational Graphs for Image Analysis," *in Proc. 7th Int'l. Conf. Pattern Recognition*, Montreal, Canada, July 30-August 3, 1984.

[119] M. Kass and A. Witkin, "Analyzing Oriented Patterns," *Computer Vision, Graphics and Image Processing*, Vol 37, No. 4, pp. 362-385, 1987.

[120] M. Kawagoe and A. Tojo, "Fingerprint Pattern Classification," *Pattern Recognition*, Vol. 17, No. 3, pp. 295-303, 1984.

[121] M. Michael and W.-C. Lin, "Experimental Study of Information Measure and Inter-Intra Class Distance Ratios on Feature Selection and Ordering", *IEEE Trans. Systems, Man, and Cybernetics*, Vol. SMC-3, No. 2, pp. 172-181, 1973.

[122] M. M. S. Chong, T. H. Ngee, L. Jun, and R. K. L. Gay, "Geometric framework for Fingerprint Classification," *Pattern Recognition*, Vol. 30, No. 9, pp. 1475-1488, 1997.

[123] M. R. Stiles, "Goverment's post-Daubert Hearing Memorandum," United States District Court for the Eastern District of Pennsylvania, USA vs Mitchell, Criminal case No. 96-00407, http://www.usao-edpa.com/ Invest/Mitchell/704postd.htm, 2000.

[124] M. R. Verma, A. K. Majumdar, and B. Chatterjee, "Edge Detection in Fingerprints," *Pattern Recognition*, vol. 20, no. 5, pp. 513-523, 1987.

[125] M. Trauring, "Automatic Comparison of Finger-ridge Patterns", *Nature*, pp. 938-940, 1963.

[126] M. Tuceryan and A. K. Jain, "Texture Analysis," in *Handbook of Pattern Recognition and Computer Vision*, C. H. Chen, L. F. Pau and P. Wang (editors), World Scientific Publishing Co., pp. 235-276, 1993.

[127] NCSA HTTPD Mosaic User Authentication Tutorial. http://hoohoo.ncsa.uiuc.edu/docs/tutorials/user.html

[128] N. Duta, A. K. Jain, and M-P. Dubuisson-Jolly, "Automatic Construction of 2D Shape Models", *IEEE Trans. Patt. Anal. and Machine Intell.* Vol. 23, No. 5, May 2001.

[129] N. L. Segal, *Entwined Lives: Twins and What They Tell Us About Human Behavior*, Plume, New York, 2000.

[130] N. Ratha, J. H. Connell, and R. M. Bolle, "An Analysis of Minutiae Matching Strength", *Proc. 3rd International Conference on Audio- and Video-Based Biometric Person Authentication*, pp. 223-228, Sweden, June 6-8, 2001.

[131] N. Ratha, K. Karu, S. Chen, and A. K. Jain, "A Real-Time Matching System for Large fingerprint Databases," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 18, No. 8, pp. 799-813, 1996.

[132] N. Ratha, Shaoyun Chen, and A. K. Jain, "Adaptive Flow Orientation-Based Feature Extraction in Fingerprint Images," *Pattern Recognition*, Vol. 28, No. 11, pp. 1657-1672, 1995.

[133] Online VoiceGuardian. http://www.keyware.com/Demos/index.html.

[134] Problem Idents. http://onin.com/fp/problemidents.html.

[135] P. Sinha and J. Mao, "Combining Multiple OCRs for Optimizing Word Recognition", *Proc. 14th Int'l Conference on Pattern Recognition*, Brisbane, pp. 436-438, Vol. 1, 1998.

[136] Q. Xiao and H. Raafat, "Fingerprint Image Postprocessing: A Combined Statistical and Structural Approach," *Pattern Recognition*, vol. 24, no. 10, pp. 985-992, 1991.

[137] R. A. Marsh and G. S. Petty, "Optical Fingerprint Correlator", *US Patent 5050220*, 1991.

[138] R. Bright, *Smartcards: Principles, Practice, Applications*, New York: Ellis Horwood, Ltd., 1988.

[139] R. Brunelli and D. Falavigna, "Person Identification Using Multiple Cues," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 17, No. 10, pp. 955-966, October 1995.

[140] R. Cappelli, A. Erol, D. Maio, and D. Maltoni, "Synthetic Fingerprint-image Generation", Proc. International Conference on Pattern Recognition (ICPR), Barcelona, Vol. 3, pp. 475-478, September 2000.

[141] R. Cappelli, D. Maio, and D. Maltoni, "Fingerprint Classification based on Multi-space KL", *Proc. Workshop on Automatic Identification Advances Technologies (AutoID'99)*, Summit (NJ), pp. 117-120, October 1999.

[142] R. Cappelli, D. Maio and D. Maltoni, "Modelling Plastic Distortion in Fingerprint Images", Proc. Second International Conference on Advances in Pattern Recognition (ICAPR2001), Rio de Janeiro, pp. 369-376, March 2001.

[143] R. Cappelli, D. Maio, and D. Maltoni, "Combining Fingerprint Classifiers", *First International Workshop on Multiple Classifier Systems (MCS2000)*, Cagliari, pp.351-361, June 2000.

[144] R. Collobert and S. Bengio, "SVMTorch: Support Vector Machines for Large-Scale Regression Problems", *Journal of Machine Learning Research*, Vol 1, pp. 143-160, 2001.

[145] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd Edition, John Wiley & Sons, November 2000.

[146] R. G. Steen, *DNA and Destiny: Nature and Nurture in Human Behavior*, New York: Plenum Press, 1996.

[147] R. M. Bolle, S. Pankanti, and Y.-S. Yao, *System and Method for Determining the Quality of Fingerprint Images*, US Patent US5963656.

[148] R. P. Brent, *Algorithms for Minimization Without Derivatives*, Engelwood Cliffs, NJ: Prentice Hall, 1973.

[149] R. Zunkel, "Hand geometry based verification", *Biometrics: Personal Identification in Networked Society*, Anil K. Jain, R. Bolle, and S. Pankanti, editors, Kluwer Academic Publishers, 1999.

[150] S. A. Cole, "What Counts for Identity?" *Fingerprint Whorld*, Vol. 27, No. 103, pp. 7-35, 2001.

[151] S. B. Meagher, B. Buldowle, and D. Ziesig, "50K Fingerprint Comparison Test", United States of America vs. Byron Mitchell, U.S. District Court Eastern District of Philadelphia. Government Exhibits 6-8 and 6-9 in Daubert Hearing before Judge J. Curtis Joyner, July 8-9, 1999.

[152] S. Chen and A. K. Jain, "A Fingerprint Matching Algorithm Using Dynamic Programming", *Technical Report*, Department of Computer Science and Engineering, Michigan State University, 1999.

[153] Secure Web Access Control: MistyGuard (TRUSTWEB). http://www.mitsubishi.com/ghp_japan/misty/trustweb_e.htm.

[154] S. E. Fahlman, "Faster-Learning Variations on Back-Propagation: An Empirical Study," *Proceedings of 1988 Connectionist Models Summer School*, 1988.

[155] S. Gold and A. Rangarajan, "A Graduated Assignment Algorithm for Graph Matching," *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 18, No. 4, pp. 377-388, 1996.

[156] Siemens ID Mouse. Available at: www.siemens.com

[157] S. L. Sclove, "The Occurrence of Fingerprint Characteristics as a Two Dimensional Process", *Journal of American Statistical Association*, Vol. 74, No. 367, pp. 588-595, 1979.

[158] S. Prabhakar and A. K. Jain, "Decision-level Fusion in Fingerprint Verification" to appear in *Pattern Recognition*, 2001.

[159] S. Prabhakar, A. K. Jain, J. Wang, S. Pankanti, and R. Bolle, "Minutiae Verification and Classification for Fingerprint Matching", *Proc. 15th International Conference on Pattern Recognition (ICPR)*, Vol. I, pp. 25-29, Barcelona, September 3-8, 2000.

[160] S. Raudys and A. K. Jain, "Small Sample Size Effects in Statistical Pattern Recognition: Recommendations for Practitioners", *IEEE Trans. on Patt. Anal. and Machine Intell.*, Vol. 13, No. 3, pp. 252-264, 1991.

[161] S. R. Gupta, "Statistical Survey of Ridge Characteristics", *Int. Criminal Police Review*, Vol. 218, No. 130, 1968.

[162] T. Chang, "Texture Analysis of Digitized Fingerprints for Singularity Detection," In *Proc. 5th International Conference on Pattern Recognition*, pp. 478-480, 1980.

[163] Thomson CSF. http://www.tcs.thomson-csf.com/fingerchip/FC_home.htm.

[164] T. K. Ho, J. J. Hull, and S. N. Srihari, "On Multiple Classifier Systems for Pattern Recognition", *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 16, No. 1, pp. 66-75, 1994.

[165] T. Kohonen, J. Kangas, J. Laaksonen, and K. Torkkola, "LVQ_PAK: A Program Package for the Correct Application of Learning Vector Quantization Algorithms," in *Proc. Intl' Joint Conf. on Neural Networks,* (Baltimore), pp. 1725-1730, June 1992.

[166] T. M. Cover, "The Best Two Independent Measurements Are Not The Two Best," *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. SMC-4, No. 1, pp. 116-117, 1974.

[167] T. M. Cover, "On the Possible Ordering in the Measurement Selection Problem", *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. SMC-7, No. 9, pp. 657-661, 1977.

[168] T. Reed, D. Carmelli, and R. H. Rosenman, "Effects of Placentation on Selected Type A Behaviors in Adult Males, in the National Heart, Lung, and Blood Institute (NHLBI) Twin Study," *Behavior Genetics*, pp. 9-19, Vol. 21, 1991.

[169] T. Reed and R. Meier, "Taking Dermatogyphic Prints: A Self-instruction Manual," *American Dermatoglyphics Association Newsletter: Supplement*, pp. 18, Vol. 9, 1990.

[170] T. Roxburgh, "On Evidential Value of Fingerprints", Sankhya: Indian Journal of Statistics, Vol. 1, pp. 189-214, 1933.

[171] U. Dieckmann, P. Plankensteiner, and T. Wagner, "Sesam: a Biometric Person Identification System Using Sensor Fusion," *Pattern Recognition Letters*, Vol. 18, No. 9, pp. 827-833, 1997.

[172] United Kingdom Biometric Working Group, "Best Practices in Testing and Reporting Biometric Device Performance", Version 1.0, March 2000. http://www.afb.org.uk/bwg/bestprac10.pdf

[173] Unpublished 1995 report by Frank Torpay of Mitre Corporation using data extracted from the FBI's Identification Division Automated Services database of $22,000,000$ human-classified fingerprints.

[174] U.S. Department of Justice document SL000386, March 2000. Online: http://www.forensic-evidence.com/site/ID/ID_fpValidation.html

[175] U.S. v. Byron Mitchell, Criminal Action No. 96-407, U.S. District Court for the Eastern District of Pennsylvania.

[176] V. Balthazard, "De l'identification par les empreintes ditalis", *Comptes Rendus, des Academies des Sciences*, No. 152, Vol. 1862, 1911.

[177] Veridicom products. Available at: www.veridicom.com

[178] W. Bodmer and R. McKie, *The Book of Man: The Quest to Discover our Genetic Heritage*, Viking, 1994.

[179] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C (2nd Ed.)*. Cambridge University Press, 1992.

[180] X. Quinghan and B. Zhaoqi, "An Approach to Fingerprint Identification by Using the Attributes of Feature Lines of Fingerprints," *Proc. Eighth Int. Conf. Pattern Recognition*, pp. 663-665, Oct. 1986.

[181] X. Jiang, W. Y. Yau and W. Ser, "Minutiae Extraction by Adaptive Tracing the Gray Level Ridge of the Fingerprint Image", *IEEE International Conference on Image Processing*, Japan, 1999.

[182] X. Jaing, W. Y. Yau, "Fingerprint Minutiae Matching based on the Local and Global Structures," *Proc. 15th International Confererence on Pattern Recognition*, Vol. 2, pp. 10421045, Barcelona, Spain, September 2000.

[183] Y. A. Zuen and S. K. Ivanov, "The Voting as a Way to Increase the Decision Reliability." *Proc. Foundations of Information/Decision fusion with applications to engineering problems*, pp. 206-210, Washington, D.C., August 1996.

[184] Y. S. Huang and C. Y. Suen, "A Method of Combining Multiple Experts for the Recognition of Unconstrained Handwritten Numerals", *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 17, No. 1, pp. 90-94, 1994.

[185] Y. Yao, P. Frasconi, and M. Pontil, "Fingerprint Classification with Combination of Support Vector Machines", *Proc. 3rd International Confernence on Audio- and Video-Based Biometric Person Authentication*, pp. 253-258, Sweden, June 6-8, 2001.

[186] Z. M. Kovacs-Vajna, "A Fingerprint Verification System Based on Triangular Matching and Dynamic Time Warping," *IEEE Trans. on Pattern Anal. and Machine Intell.*, Vol. 22, No. 11, 2000.