



Davide Maltoni
Dario Maio · Anil K. Jain
Jianjiang Feng

Handbook of Fingerprint Recognition

Third Edition

MOREMEDIA 

 Springer

Handbook of Fingerprint Recognition

Davide Maltoni · Dario Maio · Anil K. Jain ·
Jianjiang Feng

Handbook of Fingerprint Recognition

Third Edition



Davide Maltoni 
Department of Computer Science
and Engineering
University of Bologna
Cesena, Italy

Anil K. Jain
Department of Computer Science
and Engineering
Michigan State University
East Lansing, MI, USA

Dario Maio 
Department of Computer Science
and Engineering
University of Bologna
Cesena, Italy

Jianjiang Feng
Department of Automation
Tsinghua University
Beijing, China

ISBN 978-3-030-83623-8 ISBN 978-3-030-83624-5 (eBook)
<https://doi.org/10.1007/978-3-030-83624-5>

1st edition: © Springer Science+Business Media New York 2003

2nd edition: © Springer-Verlag London Limited 2009

3rd edition: © Springer Nature Switzerland AG 2022

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Overview

Biometric recognition, or simply biometrics, refers to the use of distinctive anatomical and/or behavioral characteristics or identifiers (e.g., fingerprints, face, iris, voice, and hand geometry) for automatically recognizing a person. Questions such as “Is this person authorized to enter the facility?”, “Is this individual entitled to access the privileged information?”, and “Did this person previously apply for a passport?” are routinely asked in a variety of organizations in both public and private sectors. Traditional person recognition systems that are based on ID documents and password/PIN no longer suffice to verify a person’s identity. Because biometric identifiers cannot be easily misplaced, forged, or shared, they are considered more reliable for person recognition than traditional token- (e.g., keys or ID cards) or knowledge- (e.g., password or PIN) based methods. Biometric recognition provides better security, higher efficiency, and, in many instances, offers better user convenience. It is for these reasons that biometric recognition systems have been deployed in a large number of government (e.g., border crossing, national ID card, and social benefit programs) and routine access control (e.g., mobile phone unlocking, computer logon, and access to buildings) applications.

A number of biometric recognition technologies have been developed, and several of them have been successfully deployed. Among these, fingerprints, face, and iris are the most commonly used. Each biometric trait has its strengths and weaknesses and the choice of a particular trait typically depends on the requirements of the application. Various biometric identifiers can be compared on a number of factors, including universality, distinctiveness, permanence, collectability, performance, acceptability, and circumvention. Because of the well-known distinctiveness (individuality) and persistence properties of fingerprints as well as the cost and maturity of sensors and matching algorithms, fingerprints are the most widely deployed biometric characteristics in use today. It is generally believed that the friction ridge pattern, composed of ridges and valleys, on each finger is unique. Given that there are about 7.8 billion living people on Earth and assuming each person has 10 fingers, there are 78 billion unique fingers! Hence, the biometric recognition

problem is the largest machine learning problem in terms of the number of classes. Fingerprints were first introduced as a method for person identification over 100 years ago. Now, every forensics and law enforcement agency worldwide routinely uses Automated Fingerprint Identification Systems (AFIS). While law enforcement agencies were the earliest adopters of the fingerprint recognition technology, increasing concerns about national security, financial fraud, and identity fraud have created a growing need for fingerprint technology for person recognition in a number of routine non-forensic applications.

Fingerprint recognition can be viewed as a pattern recognition system. Designing algorithms capable of extracting salient features from fingerprints and matching them in a robust way is not trivial. There is a popular misconception that automated fingerprint recognition is a fully solved problem since automated fingerprint systems have been around for almost 50 years. On the contrary, fingerprint recognition is still a challenging and stimulating recognition problem. This is particularly so when the users are uncooperative, the finger surface is dirty or scarred and the resulting fingerprint image quality is poor, and/or only small fingerprint fragments are available (e.g., due to a small sensing device). Latent fingerprints, obtained at crime scenes during forensics investigations, and their fully automated processing constitute a particularly challenging use case.

Deep learning (whose resurgence began around 2012) was a game-changer for computer vision and machine learning. The current state of the art for most biometric modalities can be attributed to the use of deep neural networks along with large training sets. Fingerprint recognition has also been approached in terms of data-driven learning techniques (as opposed to pre-specified minutiae features). This has resulted in new effective methods for automated processing of latent fingerprints and learning robust fixed-length fingerprint representations. However, top-down minutiae-based “geometric” matching still remains the best performing approach for most use cases of fingerprint recognition. This shows that tiny ridge details, first introduced for person recognition by Sir Francis Galton more than a century ago, are still competitive with the powerful representations learned by huge neural networks trained on big data.

This book reflects the progress made in automated techniques for fingerprint recognition over the past five decades, including recent deep learning-based methods. We have attempted to organize, classify, and present hundreds of existing approaches that span the end-to-end processing of fingerprints, from fingerprint sensing to final matching results in a systematic way. We hope this book would be of value to researchers interested in making contributions to this area, and system integrators and experts in different application domains who desire to explore not only the general concepts but also the intricate details of the fascinating technology behind fingerprint recognition.

Objectives

The aims and objectives of this book are to

- Introduce automated techniques for fingerprint recognition. Introductory material is provided on all components/modules of a fingerprint recognition system.
- Provide an in-depth survey of the state of the art in fingerprint recognition.
- Present in detail recent advances in fingerprint recognition, including sensing, feature extraction, matching and indexing (filtering) techniques, latent fingerprint recognition, synthetic fingerprint generation, fingerprint individuality, and design of secure fingerprint systems.
- Provide a comprehensive reference book on fingerprint recognition, including an exhaustive bibliography.

Organization and Features

After an introductory chapter, the book chapters are organized logically into four parts: fingerprint sensing (Chap. 2); fingerprint representation, matching, and classification/indexing (Chaps. 3, 4, and 5); advanced topics including latent fingerprint recognition, synthetic fingerprint generation, and fingerprint individuality (Chaps. 6, 7, and 8); and fingerprint system security (Chap. 9).

Chapter 1 introduces biometric and fingerprint systems and provides some historical remarks on fingerprints and their adoption in forensic and civilian recognition applications. All the topics that are covered in detail in the successive chapters are introduced here in brief. This will provide the reader an overview of the various book chapters and let her choose a personalized reading path. Other non-technical but important topics such as “applications” and “privacy issues” are also discussed. Some background in image processing, pattern recognition, and machine learning techniques is necessary to fully understand the majority of the book chapters. To facilitate readers who do not have this background, references to basic readings on various topics are provided at the end of Chap. 1.

Chapter 2 surveys the existing fingerprint acquisition techniques: from the traditional “ink technique” to live-scan sensing based on optical, capacitive, thermal, and ultrasonic technologies. The chapter also discusses the factors that determine the quality of a fingerprint image and introduces the technological advancements that enabled the in-display integration of fingerprint sensors in mobile phones.

Chapters 3–5 provide an in-depth treatment of fingerprint feature extraction, matching, and classification/indexing, respectively. Existing techniques are divided into various categories to guide the reader through the large number of approaches proposed in the

literature. The main approaches are explained in detail to help beginners and practitioners in the field to understand the methodology used in building fingerprint systems.

Chapters 6–8 are specifically dedicated to the three cutting-edge topics: latent fingerprint recognition, synthetic fingerprint generation, and fingerprint individuality, respectively. Deep learning methods enabled the automated processing of latent fingerprints a reality, leading to the development of a new generation of AFIS. Synthetic fingerprints have been accepted as a reasonable substitute for real fingerprints for the design, training, and benchmarking of fingerprint recognition algorithms; this approach is particularly useful to deal with emerging restrictions (e.g., European Union General Data Protection Regulation (GDPR)) on the use of Personally Identifiable Information (PII) which is defined as any data that could potentially identify a specific individual. Scientific evidence supporting fingerprint individuality is being increasingly demanded, particularly in forensic applications, and this has generated interest in designing accurate fingerprint individuality models.

Finally, Chap. 9 discusses the security issues and countermeasure techniques that are useful in building secure fingerprint recognition systems.

From the Second to the Third Edition

The third edition of the “Handbook of Fingerprint Recognition” is a major update of the second edition published in 2009. While the overall chapter structure has been mostly maintained, in the last 13 years (2009–2021) significant scientific and technological improvements have been made and this motivated us to update our manuscript to best reflect them.

The team of authors no longer includes Salil Prabhakar, now a full-time entrepreneur in the biometric business, but was enriched with the entrance of Jianjiang Feng whose scientific contributions to fingerprint recognition are well-known. A new chapter on latent fingerprint recognition was added (Chap. 6) because most advances have been made on this topic in the last decade, and it still remains a challenging problem. On the other hand, the “Biometric Fusion” chapter was removed because today multimodal biometric is mainstream, and a comprehensive introduction to this topic is already available (see Sect. 1.17).

The presentation style throughout the book chapters has also slightly changed. In the previous editions, we tried to organize and generalize the underlying ideas of all the approaches published in the literature (including minor contributions). However, with the constantly increasing number of papers appearing in journals and conferences, sticking to full coverage of the literature would lead to an over-chaotic and difficult-to-read essay. Hence, in this new edition, we have tried to balance at best a survey style with focused depth on the contributions we believe constitute major advancements.

The total length of the handbook grew from 494 to 523 pages and about 500 new papers have been cited to cover the period 2009–2021. Several new figures, drawings, and tables have been added with the aim of making the presentation illustrative and lucid. The Electronic Supplementary Material (ESM) included with the book contains the databases used in 2000, 2002, and 2004 Fingerprint Verification Competition (FVC2004). Table 1 summarizes the new content included in this edition of the Handbook.

Table 1 New content included in the Handbook

Chapter	New Content
1 Introduction	<ul style="list-style-type: none"> – Emerging applications and large-scale projects – Updated introduction to individual book chapters
2 Fingerprint sensing	<ul style="list-style-type: none"> – Evolution of sensing technology: from bulky optical devices to in-display integration in mobile phones. – New sensing technologies (e.g., OCT and touchless “on the fly”) – From CMOS to TFT sensors – Examples of multi-fingers and single-finger scanners, sensing elements for mobile devices
3 Fingerprint Analysis and Representation	<ul style="list-style-type: none"> – Advanced segmentation techniques: total variation and deep learning models – Learning-based techniques for local orientation estimation: from dictionaries to CNN – New algorithms for fingerprint pose estimation – Neural network-based enhancement of low-quality fingerprints – Learning-based minutiae detection (e.g., FingerNet) – Benchmarking local orientation estimation and minutiae detection – Pore detection with classical and deep learning approaches – Novel approaches for global and local quality computation (e.g., NFIQ2)
4 Fingerprint Matching	<ul style="list-style-type: none"> – Updated categorization of global minutiae-matching approaches – Spectral minutiae representation – Evolution of local minutiae matching: from early methods to rich local descriptors to Minutiae Cylinder Code (MCC) – Improved techniques for distortion correction – Dense fingerprint registration – Evolution of feature-based matching: from FingerCode to handcrafted textural features to deep features – DeepPrint: combining fingerprint domain knowledge with deep networks to derive compact fixed-length fingerprint representations – Pore matching – Updated overview of benchmarks and evaluation campaigns.

(continued)

Table 1 (continued)

Chapter	New Content
5 Fingerprint Classification and Indexing	<ul style="list-style-type: none"> – Shortened the sections on exclusive classification techniques and expanded fingerprint indexing and retrieval – Novel minutiae-based indexing methods – Deep learning-based indexing – Benchmarking indexing techniques
6 (New) Latent Fingerprint Recognition	<ul style="list-style-type: none"> – Latent fingerprint recognition by human experts – Automated recognition: feature extraction and matching – Latent quality estimation – Performance evaluation
7 Fingerprint Synthesis	<ul style="list-style-type: none"> – Categorization of synthetic generation approaches – Generation of a master fingerprint and derivation of multiple impressions (e.g., SFINGE) – Generative models (e.g., GAN) for the direct synthesis of fingerprint images – Validation of synthetic generators and large-scale experiments
8 Individuality	<ul style="list-style-type: none"> – Empirical versus theoretical approaches – Persistence of fingerprints
9 Securing Fingerprint Systems	<ul style="list-style-type: none"> – Threat model for fingerprint systems – Methods of obtaining fingerprint data and countermeasures – Updated introduction to presentation attack instruments – State-of-the-art presentation attack detection techniques and their performance evaluation – Altered fingerprints and their detection – Novel template protection techniques (e.g., homomorphic encryption of fixed-length representations) – Challenges and open issues

Contents of the Electronic Supplementary Material (ESM)

The book includes ESM that contains the 12 fingerprint databases used in 2000, 2002, and 2004 Fingerprint Verification Competitions (FVC). The ESM also contains a demonstration version of the SFinGe software that can be used to generate synthetic fingerprint images. These real and synthetic fingerprint images will allow interested readers to evaluate various modules of their own fingerprint recognition systems and to compare their developments with state-of-the-art algorithms.

Intended Audience

This book will be useful to researchers, practicing engineers, system integrators, and students who wish to understand and/or develop fingerprint recognition systems. It would also serve as a reference book for a graduate course on biometrics. For this reason, the book is written in an informal style and the concepts are explained in simple language. A number of examples and figures are presented to visualize the concepts and methods before giving any mathematical definition. Although the core chapters on fingerprint feature extraction, matching, and classification require some background in image processing, pattern recognition, and machine learning, the introduction, sensing, and security chapters are accessible to a wider audience (e.g., developers of biometric applications, system integrators, security managers, and designers of security systems).

Cesena, Italy

Davide Maltoni

Cesena, Italy

Dario Maio

East Lansing, USA

Anil K. Jain

Beijing, China

Jianjiang Feng

July 2021

Acknowledgments A number of individuals helped in making this book a reality. Raffaele Cappelli of the University of Bologna wrote Chap. 7 on synthetic fingerprints, and Karthik Nandakumar of MBZUAI University (Abu Dhabi) wrote Chap. 9 on securing fingerprint systems.

We also thank Wayne Wheeler at Springer, for his encouragement in revising the second edition of this book.

Contents

1	Introduction	1
1.1	Introduction	1
1.2	Biometric Recognition	2
1.3	Biometric Systems	4
1.4	Comparison of Traits	8
1.5	System Errors	13
1.5.1	Reasons Behind System Errors	13
1.5.2	Capture Module Errors	14
1.5.3	Feature Extraction Module Errors	15
1.5.4	Template Creation Module Errors	15
1.5.5	Matching Module Errors	15
1.5.6	Verification Error Rates	17
1.5.7	Identification Error Rates	21
1.5.8	Presentation Attack Detection Errors	23
1.6	System Evaluation	24
1.7	Applications of Fingerprint Systems	27
1.7.1	Application Characteristics	27
1.7.2	Application Categories	29
1.7.3	Barriers to Adoption	32
1.8	History of Fingerprints	33
1.9	Formation of Fingerprints	36
1.10	Individuality and Persistence of Fingerprints	37
1.11	Fingerprint Sensing	38
1.12	Fingerprint Representation and Feature Extraction	40
1.13	Fingerprint Matching	43
1.14	Fingerprint Classification and Indexing	45
1.15	Latent Fingerprint Recognition	47
1.16	Synthetic Fingerprints	47
1.17	Biometric Fusion	48
1.18	System Integration and Administration Issues	50

1.19	Securing Fingerprint Systems	52
1.20	Privacy Issues	54
1.21	Summary and Future Prospects	57
1.22	Image Processing, Pattern Recognition, and Machine Learning Background	58
1.22.1	Image Processing Books	58
1.22.2	Pattern Recognition and Machine Learning Books	58
1.22.3	Journals	59
	References	59
2	Fingerprint Sensing	63
2.1	Introduction	63
2.2	Off-Line Fingerprint Acquisition	68
2.3	Live-Scan Fingerprint Sensing	69
2.3.1	Optical Sensors	70
2.3.2	Capacitive Sensors	76
2.3.3	Thermal Sensors	77
2.3.4	Pressure Sensors	78
2.3.5	Ultrasound Sensors	78
2.4	Swipe Sensors	79
2.5	Fingerprint Images and Their Parameters	81
2.6	Image Quality Specifications for Fingerprint Scanners	85
2.7	Operational Quality of Fingerprint Scanners	86
2.8	Examples of Fingerprint Scanners	90
2.9	Dealing with Small-Area Sensors	95
2.10	Storing and Compressing Fingerprint Images	101
2.11	Summary	103
	References	105
3	Fingerprint Analysis and Representation	115
3.1	Introduction	115
3.2	Segmentation	120
3.2.1	Segmentation Based on Handcrafted Features and Thresholding	122
3.2.2	Learning-Based Segmentation with Simple Classifiers	123
3.2.3	Total Variation Models	124
3.2.4	Deep Learning Models	125
3.3	Local Ridge Orientation Estimation	126
3.3.1	Gradient-Based Approaches	127
3.3.2	Slit- and Projection-Based Approaches	129
3.3.3	Orientation Estimation in the Frequency Domain	131
3.3.4	Orientation Image Regularization	131

3.3.5	Global Models of Ridge Orientations	133
3.3.6	Learning-Based Methods	136
3.3.7	Benchmarking Fingerprint Orientation Extraction	138
3.4	Local Ridge Frequency Estimation	140
3.5	Singularity Detection and Pose Estimation	144
3.5.1	Poincaré	144
3.5.2	Methods Based on Local Characteristics of the Orientation Image	148
3.5.3	Partitioning-Based Methods	150
3.5.4	Methods Based on a Global Model of the Orientation Image	151
3.5.5	Fingerprint Pose Estimation	152
3.6	Enhancement	157
3.6.1	Pixel-Wise Enhancement	159
3.6.2	Contextual Filtering	160
3.6.3	Multi-Resolution and Iterative Enhancement	168
3.6.4	Learning-Based Enhancement	169
3.6.5	Crease Detection and Removal	171
3.7	Minutiae Detection	172
3.7.1	Binarization-Based Methods	174
3.7.2	Direct Gray-Scale Extraction	177
3.7.3	Learning-Based Approaches	181
3.7.4	Minutiae Encoding Standards	182
3.7.5	Benchmarking Minutiae Extraction	184
3.8	Minutiae Filtering	185
3.8.1	Structural Post-Processing	185
3.8.2	Minutiae Filtering in the Gray-Scale Domain	187
3.9	Estimation of Ridge Count	189
3.10	Pore Detection	190
3.10.1	Skeletonization	191
3.10.2	Filtering	191
3.10.3	Topological Approaches	192
3.10.4	Deep Learning Methods	193
3.11	Estimation of Fingerprint Quality	193
3.11.1	Local Quality Estimation	194
3.11.2	Global Quality Estimation	195
3.11.3	NFIQ (NIST Fingerprint Image Quality)	196
3.12	Summary	199
	References	199

4 Fingerprint Matching	217
4.1 Introduction	218
4.2 Correlation-Based Techniques	223
4.3 Minutiae-Based Methods	228
4.3.1 Problem Formulation	229
4.3.2 Similarity Score	233
4.3.3 Global Minutiae Matching Approaches	234
4.3.4 Hough Transform-Based Approaches	236
4.3.5 Consensus-Based Approaches	237
4.3.6 Spectral Minutiae Representation	241
4.3.7 Minutiae Matching with Pre-Alignment	243
4.4 Global Versus Local Minutiae Matching	244
4.4.1 Archetype Methods for Nearest Neighbor-Based and Fixed Radius-Based Local Minutiae Structures	245
4.4.2 Evolution of Local Structure Matching	247
4.4.3 Minutiae Cylinder Code	251
4.4.4 Consolidation	253
4.5 Dealing with Distortion	256
4.5.1 Fingerprint Distortion Models	257
4.5.2 Tolerance Box Adaptation	260
4.5.3 Warping	260
4.5.4 Dense Registration	262
4.5.5 A-Priori Distortion Removal	264
4.6 Feature-Based Matching Techniques	266
4.6.1 Early Global Methods	267
4.6.2 Local Orientation and Frequencies	269
4.6.3 Geometrical Attributes and Spatial Relationship of the Ridge Lines	269
4.6.4 Handcrafted Textural Features	270
4.6.5 Deep Features	271
4.6.6 Pore Matching	272
4.7 Comparing the Performance of Matching Algorithms	275
4.7.1 Fingerprint Databases	275
4.7.2 Fingerprint Evaluation Campaigns	280
4.7.3 Interoperability of Fingerprint Recognition Algorithms	281
4.8 Summary	282
References	283
5 Fingerprint Classification and Indexing	299
5.1 Introduction	299
5.2 Classification	301
5.2.1 Rule-Based Approaches	305

5.2.2	Syntactic Approaches	306
5.2.3	Structural Approaches	306
5.2.4	Statistical Approaches	307
5.2.5	Neural Network-Based Approaches	309
5.2.6	Multiple Classifier-Based Approaches	310
5.2.7	Fingerprint Sub-Classification	312
5.3	Benchmarking Fingerprint Classification Techniques	313
5.3.1	Metrics	313
5.3.2	Datasets	315
5.3.3	Search Strategies for Exclusive Classification	316
5.4	Fingerprint Indexing and Retrieval	318
5.4.1	Methods Based on Orientation and Frequency Images	320
5.4.2	Methods Based on Matching Scores	320
5.4.3	Methods Based on Minutiae	321
5.4.4	Hybrid and Ensemble Methods	324
5.4.5	Deep Learning-Based Methods	324
5.5	Benchmarking Fingerprint Indexing Techniques	330
5.5.1	Metrics and Benchmarks	330
5.5.2	Comparison of Existing Approaches	331
5.6	Summary	331
	References	332
6	Latent Fingerprint Recognition	339
6.1	Introduction	339
6.2	Latent Fingerprint Recognition by Latent Examiners	342
6.2.1	ACE-V	342
6.2.2	Criticisms	343
6.2.3	Recent Advances	344
6.3	Automated Latent Fingerprint Recognition	346
6.4	Feature Extraction	348
6.4.1	Challenges	348
6.4.2	Pose Estimation	349
6.4.3	Foreground Segmentation	352
6.4.4	Local Ridge Orientation Estimation	355
6.4.5	Overlapping Fingerprint Separation	362
6.4.6	Ridge Enhancement and Minutiae Detection	363
6.4.7	Quality Estimation	367
6.5	Matching	371
6.5.1	Challenges	371
6.5.2	Latent Matching with Manually Marked Features	374
6.5.3	Latent Matching with Automatically Extracted Features	375
6.5.4	Performance Evaluation	378

6.6	Summary	379
References		380
7	Fingerprint Synthesis	385
7.1	Introduction	385
7.2	Generation of a Master Fingerprint	387
7.2.1	Fingerprint Area Generation	389
7.2.2	Orientation Image Generation	390
7.2.3	Frequency Image Generation	395
7.2.4	Ridge Pattern Generation	395
7.3	Generation of Fingerprints from a Master Fingerprint	400
7.3.1	Variation in Ridge Thickness	401
7.3.2	Fingerprint Distortion	402
7.3.3	Perturbation and Rendering	403
7.3.4	Background Generation	404
7.4	Direct Generation of Synthetic Fingerprints	407
7.5	Validation of Synthetic Generators	410
7.5.1	Ranking Difference Among Comparison Algorithms	411
7.5.2	Match Score Distributions	412
7.5.3	Fingerprint Quality Measures	412
7.5.4	Minutiae Histograms	413
7.5.5	Analysis of Multiple Features	414
7.5.6	Large Scale Experiments	415
7.6	The “SFinGe” Software	417
7.7	Summary	422
References		424
8	Fingerprint Individuality	427
8.1	Introduction	427
8.2	Theoretical Approach	430
8.2.1	Early Individuality Models	430
8.2.2	Uniform Minutiae Placement Model	438
8.2.3	Other Models	447
8.3	Empirical Approach	448
8.4	Persistence of Fingerprints	451
8.5	Summary	453
References		453
9	Securing Fingerprint Systems	457
9.1	Introduction	458
9.2	Threat Model for Fingerprint Systems	461
9.2.1	Insider Attacks	462
9.2.2	External Adversarial Attacks	464

9.3	Methods of Obtaining Fingerprint Data and Countermeasures	466
9.3.1	Lifting Latent Fingerprints	468
9.3.2	Extracting Fingerprints from High-Resolution Photos	468
9.3.3	Guessing Fingerprint Data by Hill Climbing	470
9.3.4	Stealing Fingerprint Data from the Template Database	470
9.3.5	Countermeasures for Protecting Fingerprint Data	471
9.4	Presentation Attacks	474
9.4.1	Fingerprint Spoofs	475
9.4.2	Altered Fingerprints	476
9.5	Presentation Attack Detection	479
9.5.1	Hardware-Based Approaches for Spoof Detection	479
9.5.2	Software-Based Approaches for Spoof Detection	482
9.5.3	Altered Fingerprint Detection	485
9.5.4	PAD Performance Evaluation	487
9.5.5	Challenges and Open Issues	491
9.6	Template Protection	492
9.6.1	Desired Characteristics	494
9.6.2	Template Protection Approaches	496
9.6.3	Feature Transformation	500
9.6.4	Fingerprint Cryptosystems	504
9.6.5	Feature Adaptation	506
9.6.6	Challenges and Open Issues	510
9.7	Building a Closed Fingerprint System	512
9.8	Summary	515
	References	516

Acronyms

ACER	Average Classification Error Rate
ACE-V	Analysis, Comparison, Evaluation and Verification
AD	Auxiliary Data
AES	Advanced Encryption Standard
AFIS	Automated Fingerprint Identification System
AM	Amplitude Modulation
APCER	Attack Presentation Classification Error Rate
API	Application Programming Interface
ASPP	Atrous Spatial Pyramid Pooling
ATM	Automatic Teller Machine
BPCER	Bonafide Presentation Classification Error Rate
CCTV	Closed Circuit Television
CDEFFS	Committee to Define an Extended Fingerprint Feature Set
CJIS	Criminal Justice Information Service
CMC	Cumulative Match Characteristic
CMOS	Complementary Metal Oxide Semiconductor
CNN	Convolutional Neural Network
COTS	Commercial Off-The-Shelf
CSI	Crime Scene Investigation
CTF	Contrast Transfer Function
DCT	Discrete Cosine Transformation
DET	Detection Error Tradeoff
DoG	Difference of Gaussians
DoS	Denial of Service
DPI	Dots Per Inch
DWT	Discrete Wavelet Transform
EER	Equal Error Rate
EFS	Extended Feature Set
EFTS	Electronic Fingerprint Transmission Specification
ELFT	Evaluation of Latent Fingerprint Technology

ESD	Electrostatic Discharge
FAP	Fingerprint Acquisition Profile
FAR	False Acceptance Rate
FBI	Federal Bureau of Investigation
FFT	Fast Fourier Transform
FHE	Fully Homomorphic Encryption
FM	Frequency Modulation
FMR	False Match Rate
FNIR	False Negative Identification Rate
FNMR	False Non-Match Rate
FOE	Fingerprint Orientation Extraction
FOMFE	Fingerprint Orientation Model based on 2D Fourier Expansions
FPGA	Field Programmable Gate Array
FPIR	False Positive Identification Rate
FpVTE	Fingerprint Vendor Technology Evaluation
FRR	False Rejection Rate
FTA	Failure to Acquire
FTC	Failure to Capture
FTD	Failure to Detect
FTE	Failure to Enroll
FTIR	Frustrated Total Internal Reflection
FTP	Failure to Process
FVC DB #	FVC Database #
FVC	Fingerprint Verification Competition
FVC2000	Fingerprint Verification Competition (2000 edition)
FVC2002	Fingerprint Verification Competition (2002 edition)
FVC2004	Fingerprint Verification Competition (2004 edition)
FVC2006	Fingerprint Verification Competition (2006 edition)
FVC-onGoing	Fingerprint Verification Competition on Going
GAN	Generative Adversarial Network
GLCM	Gray-Level Co-occurrence Matrix
GPU	Graphic Processing Unit
HE	Homomorphic Encryption
HMM	Hidden Markov Model
HR	Hit Rate
i.i.d.	Independent and Identically Distributed
IAFIS	Integrated Automated Fingerprint Identification System
IARPA	Intelligence Advanced Research Projects Activity
IBIA	International Biometrics Industry Association
ICP	Iterative Closest Point
ID	Identity

IEC	International Electrotechnical Commission
IQS	Image Quality Specification
ISO	International Standards Organization
JPEG	Joint Photographic Experts Group
KL	Karhunen–Loëve
LBP	Local Binary Pattern
LCD	Liquid Crystal Display
LED	Light-Emitting Diode
LFIQ	Latent Fingerprint Image Quality
LR	Likelihood Ratio
LSH	Locality-Sensitive Hashing
MCC	Minutiae Cylinder Code
MEMS	Micro-Electro-Mechanical System
MINEX	Minutiae Interoperability Exchange Test
MoC	Match-on-Card
MRF	Markov Random Field
MTF	Modulation Transfer Function
MUT	Micromachined Ultrasound Transducer
NFIQ	NIST Fingerprint Image Quality
NGI	Next Generation Identification
NIST DB #	NIST Database #
NIST PFT	NIST Proprietary Fingerprint Template program
NIST	National Institute of Standards and Technology
NMP	Non-Match Probability
NN	Nearest Neighbor
NRC	National Research Council
NV	No Value
OCT	Optical Coherence Tomography
OTP	One Time Password
PA	Presentation Attack
PAD	Presentation Attack Detection
PAI	Presentation Attack Instrument
PC	Personal Computer
PCA	Principal Component Analysis
PCASYS	Pattern-level Classification Automation SYStem
PCB	Printed Circuit Board
PDE	Partial Differential Equation
PI	Pseudonymous Identifier
PIN	Personal Identification Number
PIV	Personal Identity Verification
PMUT	Piezoelectric Micromachined Ultrasound Transducer

PPI	Pixels Per Inch
PR	Penetration Rate
RANSAC	RANdom Sample Consensus
RBF	Radial Basis Function
RBM	Restricted Boltzmann Machine
R-CNN	Region-based Convolutional Neural Network
RF	Radio Frequency
RLC	Run Length Code
RMSD	Root Mean Square Deviation
ROC	Receiver Operating Characteristic
ROI	Region of Interest
SDK	Software Development Kit
SFinGe	Synthetic Fingerprint Generator
SIFT	Scale Invariant Feature Transformation
SNR	Signal-to-Noise Ratio
SoC	System-on-a-Chip, or System-on-Card
SoD	System on Device
SPI	Serial Peripheral Interface
SPOF	Symmetric Phase Only Filter
STFT	Short-Time Fourier Transform
SVM	Support Vector Machine
TEE	Trusted Execution Environment
TFT	Thin-Film Transistor
TPS	Thin Plate Spline
TSI	Top Sharpening Index
TV	Total Variation
USB	Universal Serial Bus
VAE	Variational AutoEncoder
VEO	Value for Exclusion Only
VID	Value for Individualization
WGAN	Wasserstein Generative Adversarial Network
WSQ	Wavelet Scalar Quantization



Introduction

1

Abstract

This chapter presents an introduction to biometric and, in particular, fingerprint recognition systems and provides some historical timeline on fingerprints and their adoption in forensic and civilian recognition applications. All the topics that are covered in detail in the successive chapters are surveyed here in brief. The notation and terminology are introduced, and error rates of a biometric system are explained and formalized by defining the main performance metrics. Other relevant topics such as biometric system applications, system integration, and privacy issues are also discussed.

Keywords

Identity recognition • Verification • Identification • Biometrics • Fingerprints • Applications • Privacy • Historical timeline of fingerprints

1.1 Introduction

More than a century has passed since Alphonse Bertillon first conceived and then industriously practiced the idea of using body measurements for solving crimes (Rhodes, 1956). Just as his idea was gaining popularity, it faded into relative obscurity by a far more significant and practical discovery of the distinctiveness of the human fingerprints. In 1893, the Home Ministry Office, UK, accepted that no two individuals have the same fingerprints. Soon after this discovery, many major law enforcement departments saw the potential of fingerprints in identifying repeat offenders who used an alias, i.e., changed their names with each arrest to evade the harshest penalties reserved for recidivists in law. The law enforcement departments embraced the idea of “booking” the fingerprints of criminals at the time of arrest, so that their records are readily available for later identification. This

is how fingerprints found an application in forensics. By matching leftover fingerprint smudges (latent prints) from crime scenes to fingerprints collected during booking, authorities could determine the identity of criminals who left their partial prints at the crime scenes. The law enforcement agencies sponsored a rigorous study of fingerprints, developed scientific methods for visual matching of fingerprints, and instituted strong programs and culture for training fingerprint experts. They successfully applied the art of fingerprint recognition for nailing down the perpetrators (Scott, 1951; Lee & Gaenslen, 2012).

Despite the ingenious methods improvised to increase the efficiency of the manual approach to fingerprint indexing and matching, the ever-growing demands on fingerprint recognition quickly became overwhelming. The manual method of fingerprint indexing (based on the Henry system of classification) resulted in a highly skewed distribution of fingerprints into bins (types): most fingerprints fell into a few bins and this did not improve the search efficiency. Fingerprint training procedures were time-intensive and slow. Furthermore, demands imposed by the painstaking attention needed to visually compare two fingerprints of varied qualities, the tedium of the monotonous nature of the work, and increasing workloads due to a higher demand on fingerprint recognition services, all prompted the law enforcement agencies to initiate research into acquiring fingerprints through electronic media and automate fingerprint recognition based on the digital representation of fingerprints. These efforts lead to the development of *Automated Fingerprint Identification Systems* (AFIS) over the past five decades. Law enforcement agencies were the earliest adopters of the automated fingerprint recognition technology. More recently, however, increasing concerns about security and identity fraud have created a growing need for fingerprint and other biometric technologies for person recognition in a large number of non-forensic applications.

1.2 Biometric Recognition

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

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

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

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

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

Thanks to the imaginative and flattering depiction of fingerprint systems in nightly television crime shows (e.g., CSI: Crime Scene Investigation), the general perception is that automated fingerprint identification is a foolproof technology! This is not true. There are a number of challenging issues that need to be addressed in order to broaden the scope of the niche market for fingerprint recognition systems.

1.3 Biometric Systems

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

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

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

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

The enrollment, verification, and identification processes involved in user recognition make use of the following system modules:

- *Capture*: a digital representation of biometric characteristics needs to be sensed and captured. A biometric sensor, such as a fingerprint scanner, is one of the central pieces of a biometric capture module. The captured digital representation of the biometric characteristic is often known as a *sample*; for example, in the case of a fingerprint system, the raw digital fingerprint image captured by the fingerprint scanner is the sample. The data capture module also has the capability to enter the subject's demographic and personal data.
- *Feature extraction*: in order to facilitate matching or comparison of fingerprints, the raw digital representation (sample) is further processed by a *feature extractor* to generate a compact but expressive representation, called a *feature set*.

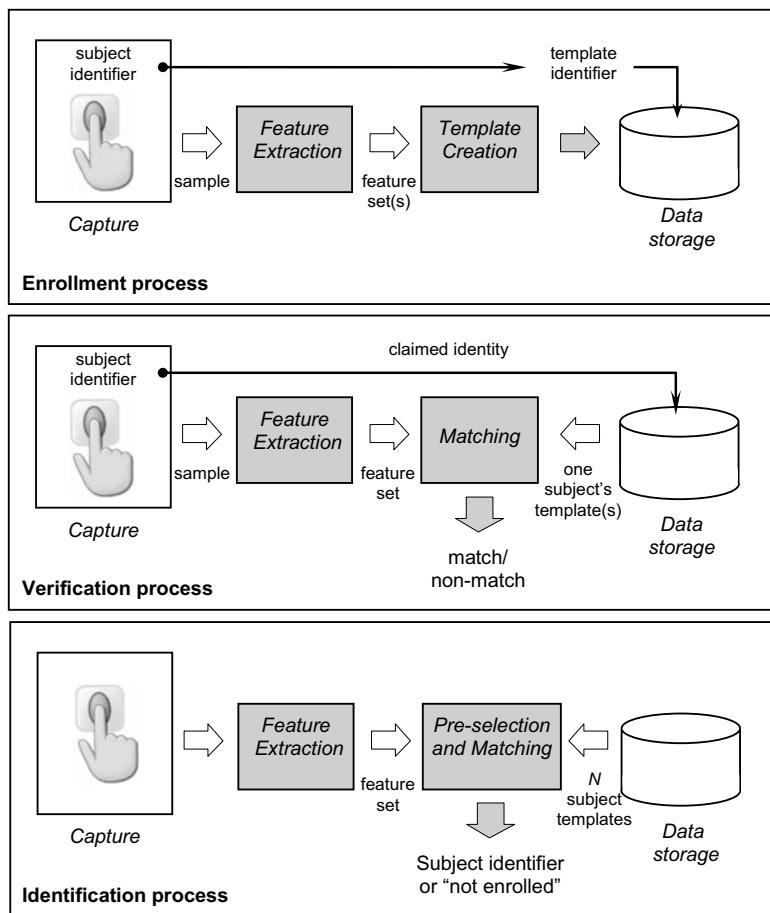


Fig. 1.1 Enrollment, verification, and identification processes. These processes use the following modules: capture, feature extraction, template creation, matching, pre-selection, and data storage. In the identification process, pre-selection and matching are often combined

- *Template creation:* the template creation module organizes one or more feature sets into an *enrollment template* that will be saved in storage media. The enrollment template is sometimes also referred to as a *reference*.
- *Pre-selection and matching:* the pre-selection (or filtering) stage is primarily used in an identification system when the number of enrolled templates in the system database is large. Its role is to reduce the effective size of the template database so that the input needs to be compared to a relatively small number of templates. The matching (or comparison) stage (also known as a *matcher*) takes a feature set and an enrollment template as inputs and computes the similarity between them in terms of a *matching score*, also known as *similarity score*. The matching score is compared to a system

threshold to make the final decision; if the match score is higher than the threshold, the person is recognized, otherwise not.

- *Data storage:* it is devoted to storing templates and other demographic information about the user. Depending on the application, the template may be stored in internal or external storage devices or be recorded on a smart card issued to the individual.

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

- *Enrollment:* user enrollment is a process that is responsible for registering individuals in the biometric system storage. During the enrollment process, the biometric characteristic of a subject is first captured by a biometric scanner to output a sample. A quality check is performed to ensure that the acquired sample can be reliably processed by successive stages. A feature extraction module is then used to produce a feature set. The template creation module uses the feature set to produce an enrollment template. Some systems collect multiple samples of a user and then either select the best image (or feature set) or fuse multiple images (or feature sets) to create a composite template. The enrollment process then takes the enrollment template and stores it in the system storage together with the demographic and other non-biometric information about the individual (such as an identifier, name, gender, and height).
- *Verification:* the verification process is responsible for confirming the claim of identity of the subject. During the recognition phase, an identifier of the subject (such as username or PIN [Personal Identification Number]) is provided (e.g., through a keypad or a proximity card) to claim an identity; the biometric scanner captures the characteristic of the subject and converts it to a sample, which is further processed by the feature extraction module to produce a feature set. The resulting feature set is fed to the matcher, where it is compared against the enrollment template(s) of that subject (retrieved from the system storage based on the subject's identifier). The verification process produces a match/non-match decision.
- *Identification:* in the identification process, the subject does not explicitly claim an identity and the system compares the feature set (extracted from the captured biometric sample) against the templates of all (or a subset of) the subjects in the system storage; the output is a *candidate list* that may be empty (if no match is found) or contain one (or more) identifier(s) of matching enrollment templates. Because identification in large databases is computationally expensive, a pre-selection stage can be used to filter the number of enrollment templates that have to be matched against the input feature set.

Depending on the application domain, a biometric system could operate either as an *online* system or an *off-line* system. An online system requires the recognition to be performed quickly and an immediate response is imposed (e.g., a mobile unlock application). On the other hand, an off-line system does not require the recognition to be performed immediately and a relatively longer response delay is allowed (e.g., background check of an applicant). Online systems are often *fully automated* and require that the biometric characteristic be captured using a live-scan scanner, the enrollment process be unattended, there be no (manual) quality control, and the matching and decision-making be fully automatic. Off-line systems, however, are often *semi-automated*, where the biometric acquisition could be through an off-line scanner (e.g., scanning a fingerprint image from a latent or inked fingerprint card), the enrollment may be supervised (e.g., when a suspect is “booked”, a police officer guides the fingerprint acquisition process), a manual quality check may be performed to ensure good-quality acquisition, and the matcher may return a list of candidates which are then manually examined by a forensic expert to arrive at a final decision.

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

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

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

The following three types of claims are used depending on the application context:

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

1.4 Comparison of Traits

Any human anatomical or behavioral trait can be used as a biometric identifier to recognize a person as long as it satisfies the following requirements:

- *Universality:* each person should possess the biometric trait.
- *Distinctiveness:* any two persons should be sufficiently different in terms of their biometric traits (or their representations).
- *Permanence:* the biometric trait should be invariant (with respect to the matching criterion) over time.
- *Collectability:* the biometric trait can be measured quantitatively.

However, in a practical biometric system, there are a number of other issues that should be considered in selecting a trait, including:

- *Performance:* recognition accuracy, speed (throughput), resource requirements, and robustness to operational and environmental factors.

- *Acceptability*: extent to which users are willing to accept the biometric identifier in their daily lives.
- *Circumvention*: ease with which the biometric system can be circumvented by fraudulent methods.

A practical biometric system should have acceptable recognition accuracy and speed with reasonable resource requirements, be harmless to the users, be accepted by the intended population, and be sufficiently robust to various fraudulent methods.

A number of biometric traits are in use in various applications. Each biometric trait has its own strengths and weaknesses and the choice typically depends on the application. No single trait is expected to effectively meet the requirements of all the applications. The match between a biometric trait and an application is determined depending upon the characteristics of the application and the properties of the trait. Some of the issues that need to be addressed in selecting a biometric trait for a particular application are as follows:

- Does the application need a verification or identification system? If an application requires an identification of a subject from a large database, it needs a very distinctive biometric trait (e.g., fingerprints or iris).
- What are the operational characteristics of the application? For example, is the application attended (semi-automated) or unattended (fully automated)? Are the users habituated (or willing to become habituated) to the given biometric? Is the application covert or overt? Are subjects cooperative or non-cooperative?
- What is the template storage requirement of the application? For example, an application that performs the recognition on a smart card may require a small template size.
- How stringent are the performance requirements? For example, an application that demands very high accuracy needs a more distinctive biometric.
- What types of biometric traits are acceptable to the target user population? Biometric traits have different degrees of acceptability in different demographic regions depending on the cultural, ethical, social, religious, and hygienic standards. The acceptability of a biometric in an application is often a compromise between the sensitivity of the targeted population to various perceptions or taboos and the value or convenience offered by biometrics-based recognition.

A brief introduction to the most common biometric traits is provided below. Corresponding visual examples are shown in Fig. 1.2. We do not cover fingerprints in this list since it is extensively covered in the rest of this book.

- *Face*: face is one of the most acceptable biometric traits and it is the most common method of recognition that humans use in their daily visual interactions. In addition, the

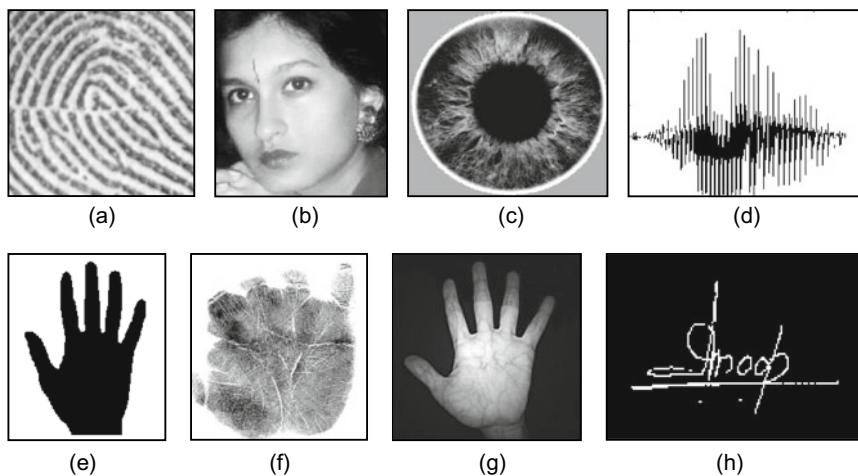


Fig. 1.2 Examples of biometrics traits: **a** fingerprint, **b** face, **c** iris, **d** voice, **e** hand geometry, **f** palmprint, **g** palm vein, and **h** signature

method of acquiring face images is nonintrusive; face can be captured remotely without your knowledge as in CCTV cameras. It is challenging to develop face recognition techniques that can tolerate the effects of aging, facial expression, variations in the imaging environment, and facial pose with respect to the camera. However, a great deal of progress has been made in this area in the last decade, thanks to deep learning models.

- *Iris:* visual texture of the human iris is determined by the chaotic morphogenetic processes during embryonic development and is posited to be distinctive for each person and each eye (Daugman, 1999). An iris image is typically captured using a noncontact imaging process. Capturing an iris image often involves cooperation from the user, both to register the image of the iris in the central imaging area and to ensure that the iris is at a predetermined distance from the focal plane of the camera. The iris recognition technology has been shown to be extremely accurate and fast on high-resolution well-captured iris images.
- *Voice:* voice may be the only feasible biometric in applications requiring person recognition over a telephone. Voice is not sufficiently distinctive to permit the identification of an individual from a large database of identities. Moreover, a voice signal available for recognition is typically degraded in quality by the microphone, communication channel, and ambient noise. Voice is also affected by factors such as a person's health (e.g., cold), stress, and emotional state. Besides, some people seem to be extraordinarily skilled in mimicking others' voices and modern voice synthesis techniques represent a big threat for automated speaker recognition.

- *Hand geometry*: some features related to the human hand (e.g., length of fingers) are relatively invariant and peculiar (although not very distinctive) to an individual. The image acquisition system requires the cooperation of the subject to capture frontal and side-view images of the palm flatly placed on a panel with outstretched fingers. The template storage requirements of the hand are very small, which is an attractive feature for bandwidth- and memory-limited systems. Due to its limited distinctiveness, hand geometry-based systems are only used for verification and do not scale well for identification applications.
- *Vein pattern*: vein-based recognition systems rely on various anatomic features like finger veins, hand veins, foot veins, or palm veins. Near-infrared (or infra-red) imaging is used to capture the vein structure. Finger vein recognition, in general, is preferable since the acquisition device is smaller, and fingers have a larger number of veins than the palm and hand.
- *Signature*: the way a person signs his name is known to be a characteristic of that individual. Signatures have been acceptable in government, legal, and commercial transactions as a method of verification for a long time. Signature is a biometric that changes over time and is influenced by the physical and emotional conditions of the signatories. Signatures of many subjects tend to vary a lot over time: even successive impressions of their signature are significantly different. Furthermore, professional forgers can reproduce signatures of others to fool the unskilled eye.

A special class of biometric traits are the so-called *soft biometrics*. They include, but are not limited to, (i) physical characteristics like skin color, eye color, hair color, presence of beard, presence of mustache, height, and weight; (ii) behavioral characteristics like gait, keystroke, and driving style, and (iii) adhered human characteristics such as clothing color, tattoos, and accessories. Soft biometrics are suitable for applications where a high degree of security is not required but usability and acceptability are the prevailing aspects. They are typically used in conjunction with a primary biometric, i.e., face, fingerprint, and iris to boost confidence in recognition.

The biometric identifiers described above are compared along several dimensions in Table 1.1. Note that fingerprint has a nice balance among all the desirable properties. Every human being possesses fingers (with the exception of hand-related disability) and hence has ten fingerprints, one per digit. Fingerprints are very distinctive (see Chap. 8), and they are permanent; even if they temporarily change slightly due to cuts and bruises on the skin, the fingerprint reappears after the finger heals. Live-scan fingerprint scanners can easily capture high-quality fingerprint images and unlike face recognition, they do not suffer from the problem of segmenting the fingerprint (ridge–valley pattern) from the background. However, they are not suitable for covert applications (e.g., surveillance) as live-scan fingerprint scanners cannot capture a fingerprint image from a distance and without the knowledge of the person. The deployed fingerprint recognition systems offer excellent recognition performance, and fingerprint scanners have become quite compact

Table 1.1 Comparison of commonly used biometric traits. Entries in the table are based on the perception of the authors. High, Medium, and Low are denoted by H, M, and L, respectively

Biometric identifier	Universality	Distinctiveness	Permanence	Collectability	Performance	Acceptability	Circumvention
Face	H	M	H	M	H	H	M
Fingerprint	M	H	M	H	M	M	M
Hand geometry	M	M	H	M	M	M	M
Iris	H	H	M	H	L	L	L
Signature	L	L	H	L	H	H	H
Vein pattern	M	M	M	M	M	L	L
Voice	M	L	M	L	H	H	H

and affordable (see Chap. 2). Because fingerprints have a long history of use in forensic divisions worldwide for criminal investigations, they used to have some stigma of criminality associated with them. However, this has rapidly changed with the high demand for automated person recognition to fight identity fraud and security threats. With a layered approach involving fingerprint and other security technologies, fingerprint systems are difficult to circumvent (see Chap. 9). Fingerprint recognition is one of the most mature biometric technologies and is suitable for a large number of recognition applications. This is also reflected in the revenues generated by various biometric technologies: according to a study prepared for the European Commission (Bonneau et al., 2018), the market is dominated by fingerprint recognition with more than 90% share. Even if we restrict focus to the consumer market (i.e., excluding government application), fingerprint still takes the dominant position with 40% of the market share, followed by face with 15% (Mordor, 2021). However, share of face recognition continues to grow due to increasing interest in surveillance.

1.5 System Errors

The critical promise of the ideal biometric trait is that when a sample is presented to the biometric system, it will offer the correct decision. In practice, a biometric system is a pattern recognition system that inevitably makes some incorrect decisions. Let us first try to understand why a biometric system makes errors and then discuss the various types of errors. We also encourage the readers to refer to ISO/IEC 19795-2 (2007) and to the other sections of ISO/IEC 19795 for a comprehensive treatment of biometric system errors.

1.5.1 Reasons Behind System Errors

There are three primary reasons that explain the errors made by a biometric system (see Jain et al., 2004):

- *Information limitation:* the invariant and distinctive information content in the biometric samples may be inherently limited due to the intrinsic signal capacity (e.g., individuality information) of the biometric identifier. For instance, the distinctive information in hand geometry is less than that in fingerprints. Consequently, hand geometry measurements can differentiate fewer identities than fingerprints even under ideal conditions. Information limitation may also be due to poorly controlled biometric presentation by the users (user interface issue) or inconsistent signal acquisition. Differently acquired measurements of a biometric identifier limit the invariance across different samples of the pattern. For example, information limitation occurs when there is very little overlap between the enrolled and sample fingerprints (e.g., left and right half of the

finger). In such situations, even a perfect matcher cannot offer a correct decision. An extreme example of information limitation is when the person does not possess or cannot present the particular biometric needed by the identification system (e.g., amputees with missing hands and fingers).

- *Representation limitation:* the ideal representation scheme should be designed to retain all the invariance as well as discriminatory information in the sensed measurements. Practical feature extraction modules, typically based on simplistic models of biometric signal, fail to capture the richness of information in a realistic biometric signal resulting in the inclusion of erroneous features and exclusion of true features. Consequently, a fraction of legitimate pattern space cannot be handled by the biometric system, resulting in errors.
- *Invariance limitation:* finally, given a representation scheme, the design of an ideal matcher should perfectly model the invariance relationship among different patterns from the same class (user), even when imaged under different presentation conditions. Again, in practice (e.g., due to the non-availability of a sufficient number of training samples, uncontrolled or unexpected variance in the collection conditions) a matcher may not correctly model the invariance relationship resulting in matcher errors.

The challenge is to be able to arrive at a realistic and invariant representation of the biometric identifier from as many samples acquired under inconsistent conditions, and then, formally estimate the discriminatory information in the signal from the samples. This is especially difficult in a large-scale identification system where the number of enrolled users is huge (e.g., in millions).

1.5.2 Capture Module Errors

In a fully automated biometric system, the biometric data is captured without human supervision and assistance. Such a biometric system typically uses a live-scan device that automatically detects the presence of a biometric characteristic as it appears in the field of view. For example, a live-scan fingerprint scanner may wait in a low-power-consumption mode, with a finger detection algorithm continually polling for the approach/presence of a finger. When the finger detection algorithm detects a finger, the scanner may switch to a finger capture mode to automatically capture a good-quality fingerprint image. The automated biometric capture system can produce two types of errors: *Failure To Detect* (FTD) and *Failure To Capture* (FTC). Failure to detect error occurs when a finger indeed approaches the fingerprint scanner, but the scanner fails to detect its presence. The failure to capture error occurs when the system knows that a finger is present but fails to capture a sample. These failures occur when either the captured image is of very poor quality (e.g., if the scanner surface is dirty) or when the capture module is used inappropriately

(e.g., only the tip of the finger instead of the central portion of the finger is presented to the scanner).

1.5.3 Feature Extraction Module Errors

After capture, the biometric sample is sent to the feature extraction module for processing. If the captured image is of poor quality, the feature extraction algorithm may fail to extract a usable feature set. This error is known as *Failure To Process* (FTP). Since capture module and feature extraction modules are common to all the processes (enrollment, verification, and identification), the three types of errors (FTD, FTC, and FTP) mentioned here are often combined into one single measure, called the *Failure To Acquire* (FTA). A high FTA rate will affect the throughput of the resulting biometric system and increase user frustration. One way to lower FTA is by increasing the sensitivity of the capture and feature extraction modules. But this will put an additional burden on the later modules (such as matching).

1.5.4 Template Creation Module Errors

The template creation module takes one (or more) feature sets extracted from samples during the enrollment process and produces a template. A failure to create a template typically happens either when there is not enough discriminatory information present in the feature sets (e.g., too small a fingerprint area) or when the fingerprint images are of poor quality and consequently the feature set(s) are very noisy. Since the template creation module is used only in the enrollment process and is the most critical part of the enrollment process, the failure of the template creation module is known as *Failure To Enroll* (FTE). There is a trade-off between the FTE rate and the error rates of the matching module discussed below. If the failure to enroll is disabled, then templates can be created from poor-quality fingerprints, but such noisy templates would result in higher matching errors.

1.5.5 Matching Module Errors

The output of a fingerprint matching module is typically a matching score (without loss of generality, lying in the interval $[0, 1]$) that quantifies the similarity between the recognition feature set and the enrollment template. The closer the score is to 1, the more certain is the system that the recognition feature set comes from the same finger that generated the enrollment template. The decision module regulates its decision by using a threshold t ; pairs of feature set and template generating scores higher than or equal to t are inferred

as *matching pairs* (i.e., belonging to the same finger), and pairs of feature set and template generating scores lower than t are inferred as *non-matching pairs* (i.e., belonging to different fingers).

When the matching module is operating in a one-to-one comparison mode (it compares feature set from one finger with a template from another finger), it gives a *match* or *non-match* decision depending on whether the comparison score exceeded the threshold or not, respectively. The matching module, operating in the one-to-one comparison mode, can commit two types of errors: (i) mistaking feature set and template from two different fingers to be from the same finger (called *false match*), and (ii) mistaking feature set and template from the same finger to be from two different fingers (called *false non-match*).

It is important to understand the difference between false match and false non-match errors and the more commonly used terminology of *false acceptance* and *false rejection* errors. The false match and false non-match are errors of the matching module in the one-to-one comparison mode, while false acceptance and false rejection are the error rates associated with verification and identification processes and in fact their exact meaning is dependent upon the type of identity claim made by the user. For example, in applications with a positive claim of identity (e.g., an access control system), a false match from the matching module results in the false acceptance of an impostor into the system, whereas a false non-match from the matching module causes the false rejection of a genuine user in the system. On the other hand, in an application with a negative claim of identity (e.g., preventing users from obtaining social benefits under false identities), a false match from the matching module results in rejecting a genuine request, whereas a false non-match from the matching module results in falsely accepting an impostor request. Further, an application may use other criteria for acceptance/rejection in addition to match/non-match decision. The notion of “false match/false non-match” is not application dependent and therefore, in principle, is more appropriate than “false acceptance/false rejection”. However, the use of false acceptance (and False Acceptance Rate, abbreviated as FAR) and false rejection (and False Rejection Rate, abbreviated as FRR) is more popular, especially in the commercial sector. In the rest of this book, while we will try to avoid the use of false acceptance and false rejection, they are synonyms for false match and false non-match, respectively.

When a biometric system operates in the identification mode, the matching module works in the one-to-many comparison mode. In its simplest form, one-to-many comparison against N templates can be viewed as a series of N one-to-one comparisons. If identification is performed only for subjects who are present in the enrollment database, the identification is known as *closed-set identification*. Closed-set identification always returns a non-empty candidate list. While closed-set identification has been studied extensively by researchers (possibly due to the convenience in performance evaluation and comparison), it is rarely used in practice. In *open-set identification*, some of the identification attempts are made by subjects who are not enrolled. In the rest of this book when we refer to identification, we will focus only on the open-set scenario. If the matching

module is given a feature set from finger A and a set of templates that includes at least one template of A, and the matching module produces an empty candidate list, the error is called a *false negative identification error*. If the matching module is given a feature set from finger A and a set of templates that does not include any template from A, and the matching module returns a non-empty candidate list, the error is called a *false positive identification error*.

1.5.6 Verification Error Rates

In the previous section, we defined the errors from the matching module when it operates in the one-to-one comparison mode that is typical of the verification process. So the false match and false non-match errors can be considered to be verification errors. Let the stored biometric template of a person be represented as \mathbf{T} and the verification feature set be represented by \mathbf{I} . Then the null and alternate hypotheses are:

$$\begin{aligned} H_0: \mathbf{I} \neq \mathbf{T}, & \text{ verification feature set does not come from the same finger as the template;} \\ H_1: \mathbf{I} = \mathbf{T}, & \text{ verification feature set comes from the same finger as the template.} \end{aligned}$$

The associated decisions are as follows:

$$\begin{aligned} D_0: & \text{non-match;} \\ D_1: & \text{match.} \end{aligned}$$

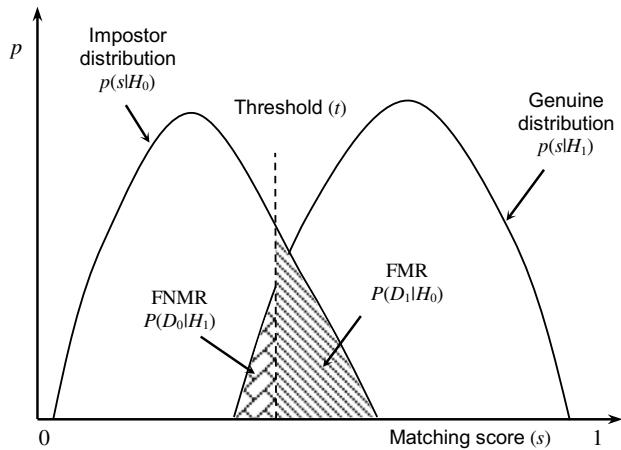
The verification involves matching \mathbf{T} and \mathbf{I} using a similarity measure $s(\mathbf{T}, \mathbf{I})$. If the matching score is less than the system threshold t , then we decide D_0 , else we decide D_1 . The above terminology is borrowed from communication theory, where the goal is to 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:

$$\begin{aligned} \text{Type I: false match } (D_1 \text{ is decided when } H_0 \text{ is true}); \\ \text{Type II: false non-match } (D_0 \text{ is decided when } H_1 \text{ is true}). \end{aligned}$$

False Match Rate (FMR) is the probability of type I error (also called significance level of the hypothesis test) and *False Non-Match Rate* (FNMR) is the probability of type II error:

$$\begin{aligned} \text{FMR} &= P(D_1|H_0); \\ \text{FNMR} &= P(D_0|H_1). \end{aligned}$$

Fig. 1.3 FMR and FNMR for a given threshold t are displayed over the genuine and impostor score distributions. Note that FMR is the percentage of impostor pairs whose comparison score is greater than or equal to t , and FNMR is the percentage of genuine pairs whose comparison score is less than t



Note that $(1 - \text{FNMR})$ is also called the power of the test.

To evaluate the accuracy of a biometric verification system, one must collect scores generated from a large number of comparisons between feature sets and enrollment templates of the same finger (the distribution $p(s|H_1)$ of such scores is traditionally called the *genuine distribution*), and scores generated from a large number of comparisons between feature sets and enrollment templates from different fingers (the distribution $p(s|H_0)$ of such scores is traditionally called the *impostor distribution*). Figure 1.3 illustrates the computation of FMR and FNMR over genuine and impostor distributions for a given threshold t :

$$\text{FNMR} = \int_0^t p(s|H_1) ds,$$

$$\text{FMR} = \int_t^1 p(s|H_0) ds.$$

There is a strict trade-off between FMR and FNMR in every biometric system (Golfarelli et al., 1997; Bazen & Veldhuis, 2004). In fact, both FMR and FNMR are functions of the system threshold t , and we should, therefore, refer to them as $\text{FMR}(t)$ and $\text{FNMR}(t)$, respectively. If t is decreased to make the system more tolerant with respect to input variations and noise, then $\text{FMR}(t)$ increases. On the other hand, if t is raised to make the system more secure, then $\text{FNMR}(t)$ increases. A system designer may not know in advance the particular application where the fingerprint system would be deployed. So it is advisable to report system performance at all operating points (threshold, t). This is done by plotting a *Receiver Operating Characteristic* (ROC) curve (or a *Detection Error Tradeoff* [DET] curve). Both the ROC and DET curves are threshold-independent allowing different fingerprint systems to be compared on a common criterion. The ROC curve

is a plot of $\text{FMR}(t)$ against $(1 - \text{FNMR}(t))$ for various decision thresholds, t . The DET curve is a plot of $\text{FMR}(t)$ against $\text{FNMR}(t)$ and provides a more direct view of the error-versus-error trade-off. Figure 1.4a–c shows examples of score distributions, $\text{FMR}(t)$ and $\text{FNMR}(t)$ curves, and a DET curve, respectively.

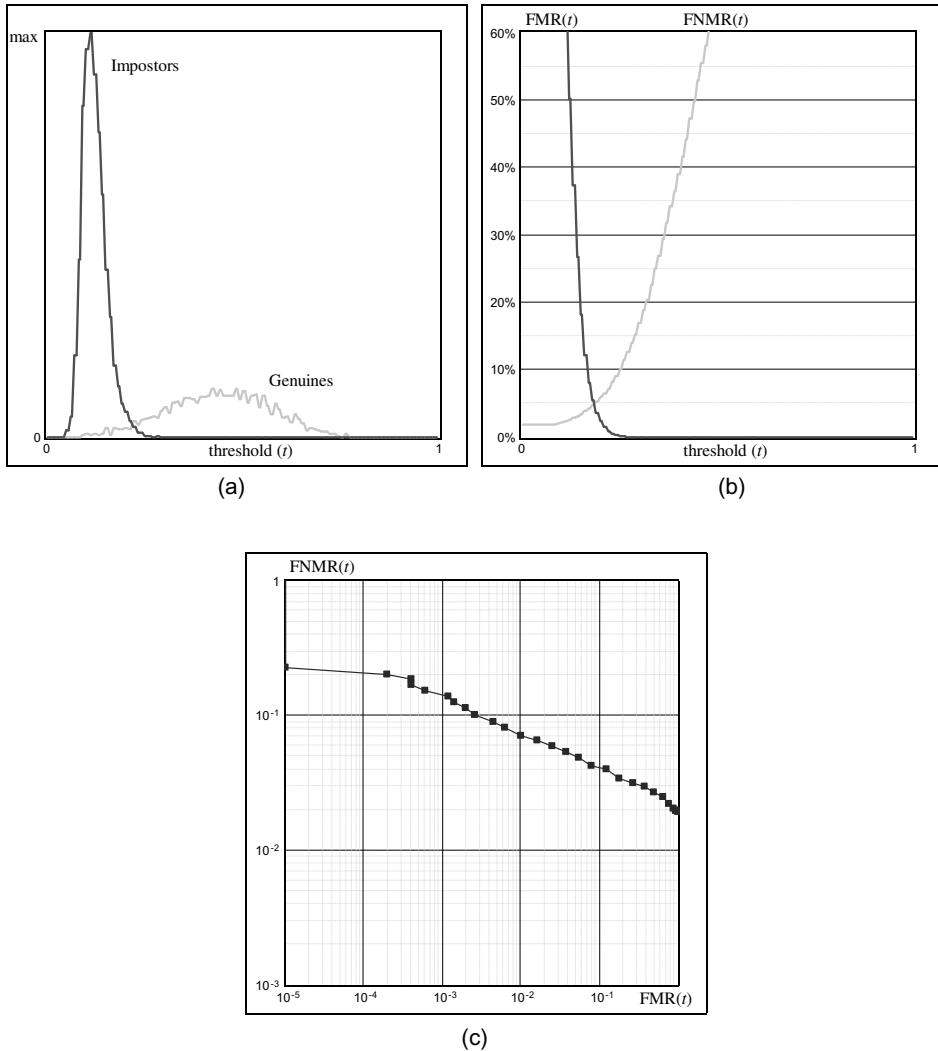
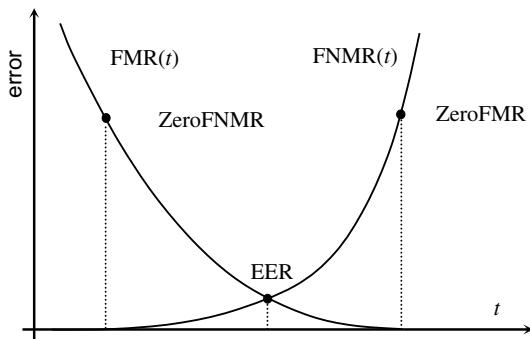


Fig. 1.4 Evaluation of a fingerprint verification algorithm over FVC2002 database DB1 (Maio et al., 2002): **a** genuine and impostor distributions were computed from 2800 genuine pairs and 4950 impostor pairs, respectively; **b** $\text{FMR}(t)$ and $\text{FNMR}(t)$ are derived from the score distributions in **a**; **c** DET curve is derived from the $\text{FMR}(t)$ and $\text{FNMR}(t)$ curves in **b**

Fig. 1.5 An example of $\text{FMR}(t)$ and $\text{FNMR}(t)$ curves, where the points corresponding to EER, ZeroFNMR, and ZeroFMR are highlighted



A few “compact” indices are sometimes used to summarize the accuracy of a verification system, but they should be used with caution.

- *Equal-Error Rate (EER)*: it denotes the error rate at the threshold t for which both false match rate and false non-match rate are identical: $\text{FMR}(t) = \text{FNMR}(t)$ (see Fig. 1.5). In practice, because the genuine and impostor score distributions are not continuous (due to the finite number of comparisons and the quantization of the output scores), an exact EER point might not exist. In this case, instead of a single value, an interval is reported (Maio et al., 2000). Although EER is an important indicator, a fingerprint system is rarely used at the operating point corresponding to EER (because the corresponding FMR is often not sufficiently low for security requirements), and often a more stringent threshold is set corresponding to a pre-specified value of FMR.
- *ZeroFNMR*: it is defined as the lowest FMR at which no false non-matches occur (see Fig. 1.5). Note that *ZeroFNMR* should be used with caution when the number of genuine matches is small.
- *ZeroFMR*: it is defined as the lowest FNMR at which no false matches occur (see Fig. 1.5). Note that the use of *ZeroFMR* should also be used with caution when the number of impostor matches is small.

For more formal definitions of errors in a fingerprint verification system, and practical suggestions on how to compute and report them for a given dataset, the reader should refer to ISO/IEC 19795-2 (2007).

The accuracy requirements of a biometric verification system are very much application-dependent. For example, in forensic applications such as criminal identification, it is the false non-match rate that is of more concern than the false match rate (which can be reduced by further manual verification): that is, we do not want to miss identifying a criminal even at the risk of manually examining a large number of potential false matches identified by the system. At the other extreme, a very low false match rate may be the most important factor in a highly secure access control application (e.g., physical access to a nuclear facility), where the primary objective is to not let in any

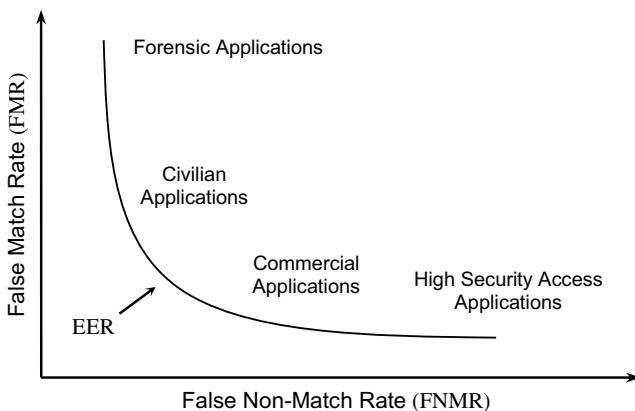


Fig. 1.6 Typical operating points of different applications displayed on an ROC curve

impostors. Of course, operating the system at a very low false match rate will potentially lead to inconvenience to legitimate users due to the resulting high false non-match rate. In between these two extremes are several commercial applications, where both false match rate and false non-match rate need to be considered. For example, in applications such as verifying a customer at a bank ATM, a false match could mean a loss of a few hundred dollars to the bank, whereas a high false non-match rate will cause inconvenience to the genuine customers. Figure 1.6 graphically depicts the FMR and FNMR trade-off preferred by different types of applications.

1.5.7 Identification Error Rates

We already defined the types of errors that the matching module can make when it operates in the one-to-many comparison mode, namely the false positive and false negative identification errors. The rates of these errors, known as *False Negative Identification-error Rate* (FNIR) and *False Positive Identification-error Rate* (FPIR) can be computed in the same way as the FNMR and FMR described above. However, they can also be interpreted from FNMR and FMR under simplifying assumptions.

Let us assume that no pre-selection algorithm is available (i.e., the entire database containing N templates is searched), that a single template for each finger is present in the database, and that all imposter matching scores are independent. Since identification error rates are dependent on the number of templates N to be searched, let us denote false negative identification-error rate as FNIR_N and false positive identification-error rate as FPIR_N . Then, under the above simplifying assumptions:

- $\text{FNIR}_N = \text{FNMR}$; the probability of false negative error when searching the identification feature set against N templates is the same as the false non-match error in verification mode (except that this expression does not take into account the probability that a false match may occur before the correct template is visited; see Cappelli et al., 2000b).
- $\text{FPIR}_N = 1 - (1 - \text{FMR})^N$; a false positive error occurs when the identification feature set falsely matches one or more templates in the database. FPIR_N is then computed as one minus the probability that no false match is made with any of the database templates. In the above expression, $(1 - \text{FMR})$ is the probability that the input does not falsely match a single template, and $(1 - \text{FMR})^N$ is the probability that it does not falsely match any of the database templates. If FMR is very small, then the above expression can be approximated by $\text{FPIR}_N \cong N \cdot \text{FMR}$, and therefore we can state that the probability of false positive errors increases linearly with the size of the database.

This expression for FPIR_N has serious implications for the design of large-scale identification systems. Usually, computation speed is perceived as the bottleneck in scaling an identification application. Actually, accuracy scales even worse than speed. Consider an identification application with 10,000 users. We can certainly find a combination of a fast matching algorithm and an appropriate platform (eventually exploiting parallelism) capable of carrying out an identification in a fraction of a second. On the other hand, suppose that, for an acceptable FNMR, the FMR of the chosen algorithm is 10^{-5} (i.e., just one false match in 100,000 matches). Then the probability of falsely matching an individual during identification is $\text{FPIR}_N \cong 10\%$, and everyone has a good chance of gaining access to the system by trying to get in by trying each of the 10 fingers in their two hands. Combining multiple fingers and other types of biometric fusion seems to be the only obvious solution to accuracy scalability in large-scale identification with tens of millions of subjects in the enrollment database.

Table 1.2 compares the state-of-the-art identification accuracy of three popular biometric identifiers.

Table 1.2 Identification accuracy reported by NIST on recent evaluation campaigns (Jain et al., 2021)

Biometric identifier	Evaluation	Gallery (N)	$\text{FNIR}@\text{FPIR} = 0.1\%$
Fingerprint (10-print fusion)	NIST FpVTE 2012	5 M	0.1%
Face	Ongoing NIST FRVT	12 M	5.8%
Iris	NIST IREX 10	500 K	0.6%

If the fingerprint templates in the database have been classified into types (see Sect. 1.12 and Chap. 5), then a pre-selection algorithm can be used such that only a portion of the database is searched during identification. This results in a different formulation of FPIR_N and FNIR_N under simplifying assumptions:

- $\text{FNIR}_N = \text{PRR} + (1 - \text{PRR}) \cdot \text{FNMR}$, where PRR (PRe-selection error Rate) is the probability that the database template corresponding to the searched finger is wrongly discarded by the pre-selection algorithm. The above expression is obtained using the following argument. In case the template is not correctly retrieved by the pre-selection algorithm (this happens with probability PRR), the system always generates a false negative error, whereas in the case where the pre-selection algorithm returns the right template (this happens with probability $[1 - \text{PRR}]$), the FNIR of the system equals FNMR. Also, this expression is only an approximation as it does not consider the probability of falsely matching an incorrect template before the right one is retrieved (Cappelli et al., 2000b).
- $\text{FPIR}_N = 1 - (1 - \text{FMR})^{N \cdot P}$, where P (also called the *Penetration rate*) is the fraction of the database searched during the identification of an input fingerprint.

A detailed analysis of errors in an identification system is derived in Cappelli et al., (2000b). The more complex case where the characteristics of the indexing and pre-selection algorithms are known is also discussed there.

The identification accuracy can also be graphically presented through an ROC curve (plotting FPIR on the x-axis and 1-FNIR on the y-axis) or a DET curve (plotting FPIR on the x-axis and FNIR on the y-axis) for a fixed database size (and fixed size of the candidate list returned). If the database is of size 1 (and the candidate list is of size 1), then the ROC and DET curves will show the accuracy performance of verification. In the special case of closed-set identification, or when interested only in the accuracy of the pre-selection algorithm, the performance can be graphically expressed using a Cumulative Match Characteristic (CMC) curve, which plots the rank (order in the candidate list) on the x-axis and the probability of identification at that or better rank on the y-axis. CMC curve is not reported as often as ROC or DET curves because closed-set identification is rarely used in practice.

1.5.8 Presentation Attack Detection Errors

A *presentation attack* is defined as the presentation of an artifact or human characteristic to the capture module with the goal of interfering with the operation of the biometric system (ISO/IEC IEC 30107-1, 2016). These attacks, discussed in Sect. 9.4, can be realized through a number of methods (e.g., a gummy finger), and the most natural countermeasure to prevent them is to add a hardware or software module to the sensor that detects

the presence of an attack. Such a module is known as Presentation Attack Detection (PAD), and is not immune to errors. Although the notation FAR/FRR is sometimes used to quantify PAD errors, the right terminology, according to ISO/IEC IEC 30107-3 (2017), is APCER/BPCER. In case of fingerprints:

- *Attack Presentation Classification Error Rate* (APCER) is the percentage of spoof fingerprints incorrectly accepted as bona fide fingerprints.
- *Bona fide Presentation Classification Error Rate* (BPCER) is the percentage of bona fide fingerprints incorrectly rejected as spoof fingerprints.

1.6 System Evaluation

Phillips et al. (2000) define three types of evaluation of biometric systems: *technology evaluation*, *scenario evaluation*, and *operational evaluation*.

- *Technology evaluation*: the goal of a technology evaluation is to compare competing algorithms from a single technology. Only algorithms compliant with a given input/output protocol are tested (sensing devices and application aspects are not taken into account). Testing of all the algorithms is carried out on one or more databases. Although sample data may be made available to prospective participants for algorithm development or parameter tuning purposes prior to the test, the actual testing must be done on data that have not previously been seen by algorithm developers. Because the test database is fixed, the results of technology tests are repeatable. Cappelli et al. (2006) propose a hierarchical taxonomy of technology evaluations: *in-house* and *independent* is the first level dichotomy. In-house evaluations can be carried out on a *self-defined test* or according to an *existing benchmark*; independent evaluations can be classified into *weakly supervised*, *supervised*, and *strongly supervised* depending on the amount of control exercised by the testing authority. Fingerprint Verification Competitions (FVC) and FVC-onGoing (see Sect. 4.7.2) are examples of strongly supervised technology evaluations of fingerprint verification algorithms. Section 4.7 discusses in more detail the technology evaluation of fingerprint recognition algorithms.
- *Scenario evaluation*: the goal of scenario evaluation is to determine the overall end-to-end system performance in a prototype or simulated application. Testing is performed on a complete system in an environment that models a real-world target application. Each tested system has its own acquisition device. Data collection across all tested systems has to be carried out in the same environment with the same population. Test results are repeatable only to the extent that the modeled scenario can be carefully controlled (ISO/IEC 19795-1, 2021).

- *Operational evaluation:* the goal of operational testing is to determine the performance of a complete biometric system in a specific application environment with a specific target population. In general, operational test results are not repeatable because of unknown and undocumented differences between operational environments (ISO/IEC 19795-1, 2021).

In scenario and operational evaluations, the accuracy of a biometric system depends heavily on several variables: the composition of the population (e.g., occupation, age, demographics, and race), the environment, the system operational mode, and other application-specific constraints. In an ideal situation, one would like to characterize the application-independent performance of a recognition system and be able to predict the real operational performance of the system based on the application. Rigorous and realistic modeling techniques characterizing data acquisition and matching processes are the only way to grasp and extrapolate the performance evaluation results. In the case of fingerprint recognition, the results of fingerprint synthesis (see Chap. 7) exhibit many characteristics of finger appearance that can be exploited for simulations, but there do not exist any formal models for the data acquisition process under different conditions (e.g., different skin conditions, different distortions, different types of cuts and their states of healing, subtle user mischief, and adversarial testing conditions). Modeling biometrics performance is a daunting task, and more effort is needed to address this problem. In the meantime, performing comparative evaluations is the norm. For example, algorithms participating in a particular FVC competition can only be compared with each other within that FVC. In other words, results from one FVC (say FVC2004) cannot be compared with results from another FVC (say FVC2006) due to the vast differences in a large number of factors such as the composition of the population (e.g., fraction of male/females, differences in occupations, and age), fingerprint scanner, demographics, ergonomics, and environment.

Until many aspects of biometric recognition algorithms and application requirements are clearly understood, the comparative, empirical, application-dependent evaluation techniques will be predominant, and the evaluation results obtained using these techniques will be meaningful only for a specific database in a specific test environment and a specific application. The disadvantage of the empirical evaluation is that it is not only expensive to collect the data for each evaluation, but it is also often difficult to objectively compare the evaluation results of two different systems tested under different conditions. Depending upon the data collection protocol, the performance results can vary significantly from one test to another. For example, in FVC2004, the fingerprint data was collected by intentionally introducing various types of finger distortion. While this is a good way to compare participating algorithms under a “stress-test”, the FVC2004 accuracy performance cannot be compared with FVC2006 where no intentional distortions were introduced. Finally, biometric samples collected in a controlled and laboratory environment provide optimistically biased results that do not generalize well in practice.

For any performance metric to be able to generalize to the entire population of interest, the test data should (i) be *representative* of the population and (ii) contain a sufficient number of samples from each category of the population (*large sample size*). Furthermore, the collection of samples for enrollment and recognition should be separated in time (e.g., 2–3 weeks or months for fingerprints). Different applications, depending on whether the subjects are cooperative and habituated, or whether the target population is benevolent or subversive, may require a completely different sample set (Wayman, 2001). The size of the sample set is a very important factor in obtaining a reliable estimate of the error rates. The larger the test sample size, the more reliable is the test result (smaller confidence interval). Data collection is expensive, so it is desirable to determine the smallest sample size that will result in a given confidence interval. Some efforts have been made to estimate the sample size (Doddington et al., 1998; Wayman, 2001; ISO/IEC 19795-1, 2021).

There are two methods of estimating confidence intervals: parametric and nonparametric. To simplify the estimation, both approaches typically assume independent and identically distributed (i.i.d.) test samples (genuine and imposter match scores). Furthermore, parametric methods make strong assumptions about the form of the (genuine and imposter) match score distributions. A typical parametric approach models the test samples as independent Bernoulli trials and estimates the confidence intervals based on the resulting binomial distribution, inasmuch as a collection of correlated Bernoulli trials is also binomially distributed with a smaller variance (Viveros et al., 1984). Similarly, non-identically distributed test samples can be accommodated within the parametric approach by making some simplifying assumptions about the data. Wayman (2001) applied these methods to obtain estimates of accuracies as well as their confidence intervals. A non-parametric approach, such as bootstrap, has been used by Bolle et al. (1999) to estimate the error rates as well as their confidence intervals. The nonparametric approaches do not make any assumption about the form of the distributions. In addition, some nonparametric approaches such as bootstrapping techniques are known to be relatively immune to violations of i.i.d. assumptions. Bolle et al. (2004) further explicitly modeled the weak dependence among typical fingerprint test sets by using a *subset bootstrap* technique. This technique obtains a better estimate of the error rate confidence intervals than the techniques that do not take the dependency among the test data into account.

In summary, the performance of a biometric system is determined empirically. The results of these evaluations should be interpreted keeping the test data collection protocol in mind. Fortunately, the biometric standards community has established the *best practices* guidelines and standards for biometric performance testing and reporting (ISO/IEC 19795-1 2021, ISO/IEC 19795-2 2007, ISO/IEC TR 19795-3 2007, and ISO/IEC 19795-4 2008).

1.7 Applications of Fingerprint Systems

Fingerprint recognition systems have been deployed in a wide variety of application domains, ranging from forensics to mobile phones. But the system design depends on the application characteristics that define the application requirements.

1.7.1 Application Characteristics

Wayman (1999) suggests that the application context of a biometric recognition system can be understood by examining the following characteristics:

1. Cooperative versus non-cooperative;
2. Habituated versus non-habituated;
3. Attended versus non-attended;
4. Standard versus non-standard operating environment;
5. Public versus private;
6. Open versus closed;
7. Overt versus covert.

Cooperative versus non-cooperative dichotomy refers to the behavior of an impostor in interacting with the fingerprint recognition application. For example, in a positive recognition application (i.e., an application that assumes a positive claim of identity), it is in the best interest of an impostor to cooperate with the system to be accepted as a valid user. On the other hand, in a negative recognition application (i.e., an application that assumes a negative claim of identity), it is in the best interest of the impostor not to cooperate with the system so that the system does not find her matching any of the individuals in the watch list. Electronic banking is an example of a cooperative application, whereas an airport application to catch terrorists is an example of a non-cooperative application.

If a subject is aware that she is being recognized using biometrics, the application is categorized as overt. If the subject is unaware, the application is covert. Facial recognition can be used in a covert application (by surveillance cameras), while fingerprint recognition cannot be used in this mode (except for criminal identification based on latent fingerprints). Most commercial applications of biometrics are overt, whereas some government and law enforcement applications are covert.

Habituated versus non-habituated use of a biometric system refers to how often the users in that application interact with the biometric recognition system. For example, a computer network logon application typically has habituated users (after an initial habituation period) due to their use of the system on a regular basis. However, in a driver's license application, the users are non-habituated since a driver's license is renewed only

once in 5 years or so. This is an important consideration when designing a biometric system because the familiarity of users with the system affects its recognition accuracy.

Attended versus non-attended classification refers to whether the process of biometric data acquisition in an application is observed, guided, or supervised by a human (e.g., a security officer). Furthermore, an application may have an attended enrollment but non-attended recognition. For example, a banking application may have a supervised enrollment when an ATM card is issued to a user, but the subsequent uses of the biometric system for ATM transactions will be non-attended. Non-cooperative applications generally require attended operation.

Standard versus non-standard environment refers to whether the application is being operated in a controlled environment (such as temperature, pressure, moisture, and lighting conditions). Typically, indoor applications such as computer network logon operate in a controlled environment, whereas outdoor applications such as parking lot surveillance operate in a non-standard environment. This classification is also important for the system designer as a more rugged biometric scanner is needed for a non-standard environment. Similarly, infrared face recognition may be preferred over visible-band face recognition for outdoor surveillance at night.

Public or private dichotomy refers to whether the users of the application are customers or employees of the organization deploying the biometric system. For example, a network logon application is used by the employees and managed by the information technology manager of that company. Thus, it is a private application. The use of biometric data in conjunction with electronic identity cards is an example of a public application.

Closed versus open application refers to whether a person's biometric template is used for a single or multiple applications. For example, a user may use a fingerprint-based recognition system to enter secure facilities, for computer network logon, electronic banking, and ATM. Should all these applications use separate template storage for each application, or should they all access the same central template storage? A closed application may be based on a proprietary template, whereas an open system may need a standard biometric data format and compression method to exchange and compare information among different systems (most likely developed by different vendors).

Note that the most popular commercial applications have the following attributes: cooperative, overt, habituated, attended enrollment and non-attended recognition, standard environment, closed, and private. A registered traveler application has the following typical attributes (Wayman, 1999): cooperative, overt, non-attended, non-habituated, standard environment, public, and closed. A driver license application (to prevent the issuance of multiple licenses to the same person) can be characterized by the following attributes: non-cooperative, overt, attended, non-habituated, standard environment, public, open application.

A fingerprint system designer can take advantage of the application characteristics to make sure that the system makes the correct trade-offs. These trade-offs may include recognition accuracy, response time, system integrity, complexity, cost (component price

as well as integration and support costs), privacy, government standards, liveness detection, ease of integration, durability, modality of usage, etc. For example, a commercial application that requires the fingerprint recognition system to work for all the people all the time demands a high recognition accuracy which may come at an expense of requiring powerful processors and large memory or specialized biometric capture equipment (for example, large-area fingerprint readers or face recognition booths with controlled lighting). In another example, compliance with certain government standards may facilitate inter-operability but may decrease recognition accuracy.

Fingerprint vendors spend a great deal of effort in optimizing and balancing the various trade-offs for the applications they target. Trade-offs in commercial applications typically include fingerprint scanner size, scanner cost, scanner ruggedness, recognition accuracy, template size, memory and cache size, security issues, system design, etc. In general, all commercial applications are typically cost-sensitive with a strong incentive for being user-friendly. On the other hand, most government applications are large-scale with a strong incentive for high data acquisition throughput.

1.7.2 Application Categories

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

- *Physical access control*: access is restricted to facilities such as nuclear plants, bank vaults, corporate board rooms, and even health clubs, amusement parks, vehicles, and lockers.
- *Logical access control*: access to desktop computers or remote servers and databases is restricted to authorized users. Increasingly, access to software applications is also being restricted to only authorized users.
- *Transaction authentication* (or consumer identification): transactions may be executed at ATM sites or from remote locations for online banking or between banks (e.g., in high-value transactions). Fingerprint recognition systems are used for the security of the transaction as well as accountability (so the parties involved in the transaction cannot later deny it).
- *Device access control*: smartphones, laptops, and other electronic devices (e.g., health monitors) often contain personal and sensitive data. To protect such data, fingerprint recognition systems are used to conduct recognition on stand-alone devices.

- *Time and attendance:* time and attendance systems are used to keep track of employee working hours and to compute payrolls. The use of fingerprint recognition systems in these applications is fairly well-received to improve efficiency for employees and also for preventing various types of payroll frauds (e.g., buddy-punching).
- *Civil identification:* in civilian identification application, the most important objective is to prevent multiple enrollments and to find duplicates (e.g., duplicate passport, driver license, and national identification card). The size of the database can be of the order of millions (e.g., the entire population of a country). In some applications (such as border control to prevent suspected terrorists or expellees from entering the country), the identification is not needed to be conducted against the entire population but rather against a “watch-list” database. The world’s largest civil registration system is India’s Aadhaar which has enrolled almost the entire population of approximately 1.4 billion (<https://uidai.gov.in/>).
- *Forensic and law enforcement identification:* in forensic identification, latent fingerprints lifted from the crime scenes are matched against a criminal database to identify the suspect (and sometimes the victims). In law enforcement, a suspect at a booking time may provide an “alias” (not his true name). To check it, the suspect’s tenprint is compared to the tenprints in the database.

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

- Health care,
- Financial,
- Gaming and hospitality (casinos, hotels, etc.),
- Retail,
- Education,
- Manufacturing,
- High technology and telecommunications,
- Travel and transport,
- Federal, state, municipal, or other governments,
- Military,
- Law enforcement, and
- Social benefit.

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

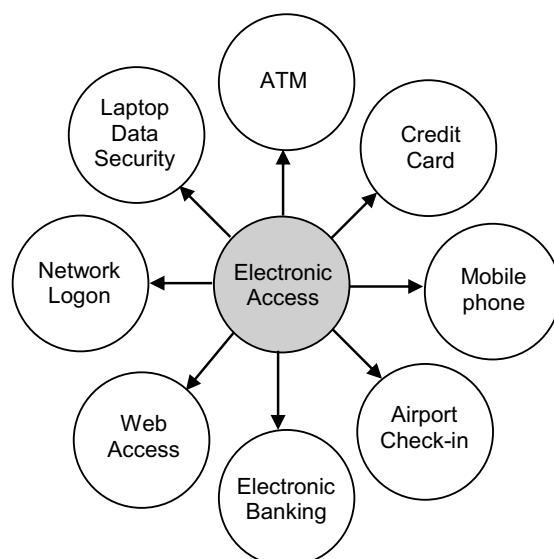
physical access control solutions. Fingerprint-based time and attendance solutions can also be used to manage the payroll of law enforcement officers (and other employees of the department).

Table 1.3 shows a categorization that lists applications that are most critical in three major vertical markets. Figure 1.7 shows some applications involving electronic access or transactions where reliable user recognition has become critical. It is not possible to list all the possible applications of fingerprint recognition systems, nor is it possible to list all the market segments. New applications continue to emerge. We are also witnessing

Table 1.3 Fingerprint recognition applications are divided here into three categories. Traditionally, forensic applications have used forensic experts, government applications have used token-based systems, and commercial applications have used knowledge-based (password) systems. Fingerprint recognition systems are now being increasingly used in all these sectors

Forensic	Government (Civil)	Commercial
Corpse Identification	Social Security	Computer Network Logon
Criminal Investigation	Civil Registration	Electronic Data Security
Missing and Exploited Children and Adults	Social Benefits Border Control Passport Control National ID card Driver License Credentialing	e-Commerce Internet Access ATM, Credit Card Physical Access Control Smartphones Personal Digital Assistant Medical Records Management Distance Learning

Fig. 1.7 Various applications involving electronic access or transaction that require reliable automatic user recognition



the maturity of products and a range of solutions. Some vendors are now offering products that address multiple horizontal applications with a single solution, for example, a single solution that can manage both the logical access and physical access control in an organization.

1.7.3 Barriers to Adoption

A fingerprint recognition system provides a good balance of security, privacy, convenience, and accountability. While the adoption of these systems is steadily increasing, the rate of adoption has been somewhat slower than anticipated. This is primarily because of a lack of awareness about the benefits and capabilities of fingerprint technologies. Another reason is that the business case for fingerprint recognition systems (return on investment analysis) has often proven to be somewhat difficult due to the following reasons:

- The business value of “security” and “deterrence” is difficult to quantify in terms of return on investment, regardless of the technology.
- Fraud rates and the resulting cost of long-standing business and government systems (for example, tokens and passwords) are not well understood and quantified.
- Fingerprint recognition systems, being a mature technology, are sometimes confronted with unrealistic performance expectations and not fairly compared with existing alternatives (for example, tokens and passwords), whose inconvenience and high-cost businesses have resigned to tolerate. A successful fingerprint-based solution does not have to be perfect in terms of accuracy and foolproof in terms of security. A particular application simply demands a satisfactory recognition performance justifying the additional investments needed for the fingerprint system. The system designer can exploit the application context to engineer the system to achieve the target performance levels at an acceptable cost.
- The requirements of available fingerprint technology vary quite dramatically from one application to another, and its performance varies from one vendor to another. Businesses cannot often easily access and understand credible reports on technology evaluations because all the vendors do not participate in standardized scenario testing of fingerprint systems. This leaves businesses to either perform their own evaluation (which delays deployment and can be expensive) or rely on references (which could be difficult to obtain because of unique operational scenarios).
- Several fingerprint system vendors are not financially stable, leaving businesses with concerns over continued product and support availability.

In the past, the most concrete return on investment estimates for businesses has come from taking people out of business processes and transactions. For example, forgotten passwords result in helpdesk calls which are expensive to businesses. It is now widely

documented that fingerprint systems can significantly reduce the spending on helpdesk calls much beyond the cost of investment. In many commercial applications, the use of fingerprint systems can facilitate businesses to move to a user-friendly self-service model of service and support while providing the same or even higher level of security as the attended model, thus lowering their expenses. In applications with a negative claim of identity, such as background check (often used for “security clearance”), voter registration, and multiple enrollments (duplicate passport and driver’s license), there are no alternatives to biometrics. Here, the main barrier to adoption has been public perception and privacy concerns. Fingerprint recognition systems, when properly implemented, provide more security, convenience, and efficiency than any other means of identification. No other technology has the capability to provide non-repudiation or ensure that the person being authenticated is physically present at the point of authentication. Fingerprint-based recognition systems have already replaced passwords and tokens in a large number of applications. In some other applications, they are used to add a layer of security on top of passwords and tokens. The use of fingerprint recognition systems will continue to reduce identity theft and fraud and protect privacy. It is clear that fingerprint-based recognition has made a profound influence on the way we conduct our daily business.

1.8 History of Fingerprints

Human fingerprints have been discovered on a large number of archeological artifacts and historical items (see Fig. 1.8). While these findings provide evidence that ancient people were aware of the individuality of fingerprints, such awareness does not appear to have any scientific basis (Lee & Gaensslen, 2012; Moenssens, 1971). It was not until the late sixteenth century that the modern scientific fingerprint technique was first initiated (see Cummins & Midlo, 1961; Galton, 1892; Lee & Gaensslen, 2012). In 1864, the English plant morphologist, Nehemiah Grew, published the first scientific paper reporting his systematic study on the ridge, furrow, and pore structure in fingerprints (Fig. 1.9a) (Lee & Gaensslen, 2012).

The first detailed description of the anatomical formation of fingerprints was made by Mayer in 1788 (Moenssens, 1971) in which a number of fingerprint ridge characteristics were identified and characterized (Fig. 1.9b). Starting in 1809, Thomas Bewick started using fingerprint as his trademark (Fig. 1.9c), one of the most important milestones in the history of fingerprints (Moenssens, 1971). Purkinje, in 1823, proposed the first fingerprint classification scheme, which classified fingerprints into nine categories according to the ridge configurations (Fig. 1.9d) (Moenssens, 1971).

Henry Fauld, in 1880, first scientifically suggested the individuality of fingerprints based on empirical observations. At the same time, Herschel asserted that he had practiced fingerprint recognition for about 20 years (Lee & Gaensslen, 2012; Moenssens, 1971). These findings established the foundation of modern fingerprint recognition.

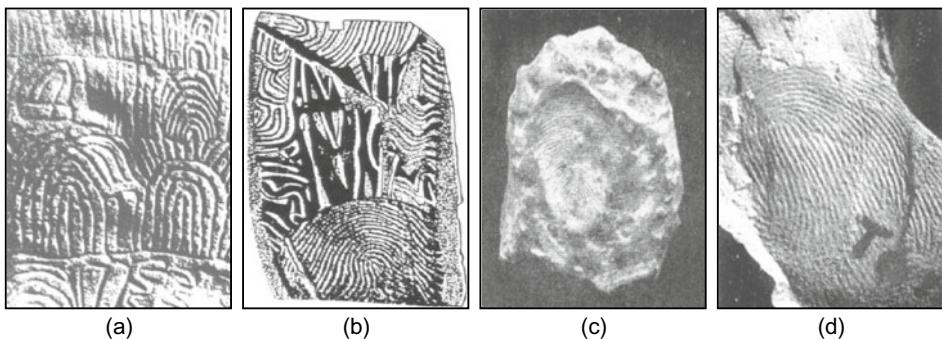


Fig. 1.8 Examples of archeological fingerprint carvings and historic fingerprint impressions: **a** Neolithic carvings (Gavrinis Island) (Moenssens, 1971); **b** standing stone (Goat Island, 2000 B.C.) (Lee & Gaensslen, 2012); **c** a Chinese clay seal (300 B.C.) (Lee & Gaensslen, 2012); **d** an impression on a Palestinian lamp (400 A.D.) (Moenssens, 1971). While impressions on the Neolithic carvings and the Goat Island standing stones might not be used to establish identity, there is sufficient evidence to suggest that the Chinese clay seal and impressions on the Palestinian lamp were used to indicate the identity of the fingerprint providers. Figure courtesy of A. Moenssens, R. Gaensslen, and J. Berry

In the late nineteenth century, Sir Francis Galton conducted an extensive study on fingerprints. He introduced the minutiae features for comparing fingerprints in 1888. In his 1892 book titled “Finger Prints” (Galton, 1892), he argued: *“They have the unique merit of retaining all their peculiarities unchanged throughout life, and afford in consequence an incomparably surer criterion of identity than any other bodily feature.”* (Engelsma et al., 2021).

An important advance in fingerprint recognition was made in 1899 by Edward Henry, who (actually his two assistants from India) established the well-known “Henry system” of fingerprint classification (Lee & Gaensslen, 2012). By the early twentieth century, the formation of fingerprints was well understood. The biological principles of fingerprints (Moenssens, 1971) are summarized as follows:

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

The first principle constitutes the foundation of fingerprint recognition, and the second principle constitutes the foundation of fingerprint classification.

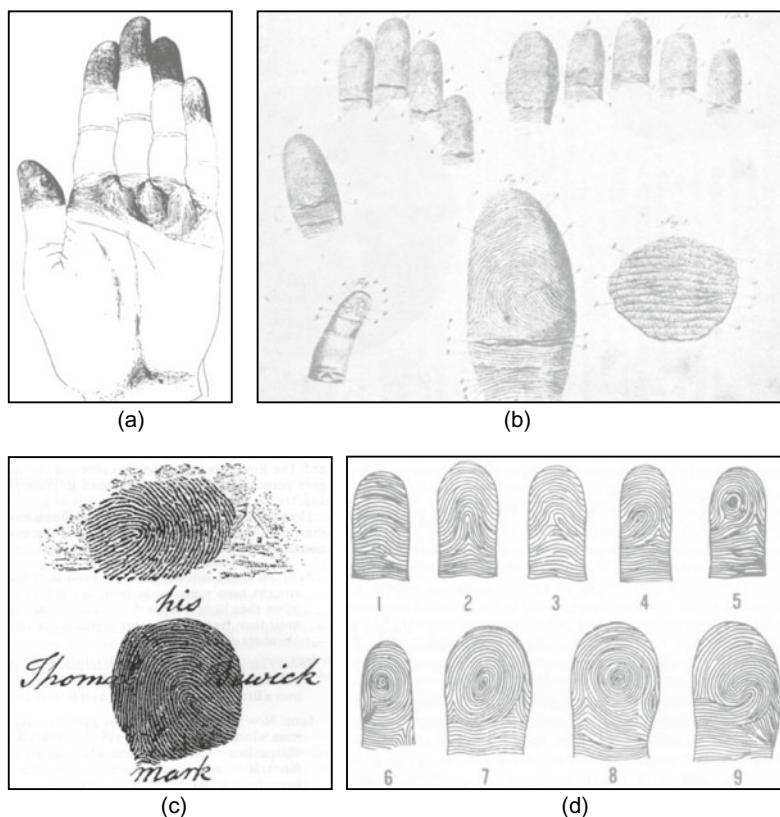


Fig. 1.9 **a** Dermatoglyphics drawn by Grew (Moenssens, 1971); **b** Mayer's drawings of fingerprints (Cummins & Midlo, 1961); **c** trademark of Thomas Bewick (Lee & Gaensslen, 2012); **d** the nine patterns illustrated in Purkinje's thesis (Moenssens, 1971). Image courtesy of A. Moenssens, R. Gaensslen, and J. Berry

In the early twentieth century, fingerprint recognition was formally accepted as a valid personal identification method and became a standard routine in forensics (Lee & Gaensslen, 2012). Fingerprint identification agencies were set up worldwide and criminal fingerprint databases were established (Lee & Gaensslen, 2012). Various fingerprint recognition techniques, including latent fingerprint acquisition, fingerprint classification, and fingerprint comparison were developed. For example, the FBI fingerprint identification division was set up in 1924 with a database of 810,000 fingerprint cards (see Federal Bureau of Investigation 1984, 1991).

With the rapid expansion of fingerprint recognition in forensics, operational fingerprint databases became so huge that manual fingerprint identification became infeasible. For example, the total number of fingerprint cards (each card contains one impression for each of the 10 fingers of a person) in the FBI fingerprint database now stands well over

150 million from its original number of 810,000 and is growing continuously (FBI, 2021). With thousands of search requests for finding a mate being received daily, even a team of more than 1,300 fingerprint experts were not able to provide timely responses to these requests (Lee & Gaenslen, 2012). Starting in the early 1960s, the FBI, Home Office in the UK, and Paris Police Department began to invest a large amount of effort in developing automated fingerprint identification systems (Lee & Gaenslen, 2012). Based on the observations of how human fingerprint experts perform fingerprint recognition, three major problems in designing AFIS were identified and investigated: digital fingerprint acquisition, local ridge characteristic extraction, and ridge characteristic pattern matching. Their efforts were so successful that today almost every law enforcement agency worldwide uses an AFIS. These systems have greatly improved the operational productivity of law enforcement agencies and reduced the cost of hiring and training human fingerprint experts. For example, in the month of April 2021 alone, the Next Generation Identification (NGI) system operated by FBI processed (i) 549,429 criminal tenprint search submissions, (ii) 2,372,774 civil fingerprint search submissions, and (iii) more than 25,000 latent search requests (FBI, 2021).

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

1.9 Formation of Fingerprints

Fingerprints are fully formed at about 7 months of fetus development. Finger ridge configurations do not change throughout the life of an individual except due to accidents such as bruises and cuts on the fingertips (Babler, 1991). This property makes fingerprints a very attractive biometric identifier. Biological organisms, in general, are the consequence of the interaction of genes and the 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. Fingerprint formation is similar to the growth of capillaries and blood vessels in angiogenesis. The general characteristics of the fingerprint emerge as the skin on the fingertip begins to differentiate. The differentiation process is triggered by the growth in the size of the volar pads on the palms, fingers, soles, and toes. 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 the 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 exactly

alike. But, because the fingerprints are differentiated from the same genes, they are not totally random patterns either.

The extent of variation in a human physical trait due to a random development process differs from trait to trait. Typically, most of the physical characteristics such as body type, voice, and face are very similar for identical twins, and automatic recognition based on face and hand geometry will most likely fail to distinguish them. Although the minute details in the fingerprints of identical twins are different (Jain et al., 2002), a number of studies have shown a significant correlation in the fingerprint class (i.e., whorl, right loop, left loop, arch, and tented arch) of identical (monozygotic) 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 be significant in identical twins (Lin et al., 1982). In dermatoglyphics studies, the maximum generic difference between fingerprints has been found among individuals of different races. Unrelated persons of the same race have very little generic similarity in their fingerprints, parent and child have some generic similarity as they share half the genes, siblings have more similarity, and the maximum generic similarity is observed in monozygotic (identical) twins, which is the closest genetic relationship (Cummins & Midlo, 1943).

1.10 Individuality and Persistence of Fingerprints

Uniqueness or individuality typically means that the friction ridge pattern in each finger is individual, while persistence typically refers to the invariance of the friction ridge pattern itself. However, the pertinent question of interest is whether or not two impressions of friction ridge skin can be attributed to the same finger, especially when the size of impressions is small and/or the quality is non-ideal. As pointed out in the National Academy of Sciences' Report, *Strengthening Forensic Science in the United States: A Path Forward*, "Uniqueness and persistence are necessary conditions for friction ridge identification to be feasible, but those conditions do not imply that anyone can reliably discern whether or not two friction ridge impressions were made by the same person." Cole (2001) argues that uniqueness may be valid when entire fingerprints are compared but not for fingerprints depicting small portions of a finger.

With the stipulation of widespread use of fingerprints, there is a rightfully growing public concern about the scientific basis underlying the individuality of fingerprints. In fact, the scientific basis of fingerprint individuality has been questioned in several court cases in the United States (see Chap. 8). To quantitatively study fingerprint individuality and persistence, researchers have proposed various theoretical or empirical methods in the past, which are discussed in Chap. 8.

1.11 Fingerprint Sensing

Based on the mode of acquisition, a fingerprint image may be classified as off-line or live-scan. An off-line image is typically obtained by smearing ink on the fingertip and creating an inked impression of the fingertip on paper. The inked impression is then digitized by scanning the paper using an optical scanner or a high-quality video camera. A live-scan image, on the other hand, is acquired by sensing the tip of the finger directly, using a sensor that is capable of digitizing the fingerprint on contact. Particular kinds of off-line images, extremely important in forensic applications, are the so-called *latent* fingerprints found at crime scenes. The oily nature of the skin results in the impression of a fingerprint being deposited on a surface that is touched by a finger. These latent prints can be “lifted” from the surface by employing certain chemical techniques. The main problem with latents is that they can often be of poor quality and, in those instances, they cannot be recaptured because we do not know who left the print.

The main parameters characterizing a digital fingerprint image are resolution, area, number of pixels, geometric accuracy, contrast, and geometric distortion. To maximize compatibility between digital fingerprint images and to ensure good quality of the acquired fingerprint impressions among various AFIS, the FBI Criminal Justice Information Services (CJIS) released a set of specifications that regulate the quality and the format of both fingerprint images and FBI-compliant off-line/live-scan scanners (Appendices F and G of CJIS, 2017). FBI also defined another, less stringent, image quality standard (CJIS, 2006) for single-finger capture devices in civilian applications (more specifically for the Personal Identity Verification [PIV] program in the United States). Most of the commercial live-scan devices, designed for the non-AFIS market, do not meet the FBI specifications but, on the other hand, are designed to be compact, and cheap. The operational quality of fingerprint scanners (i.e., the impact of the scanner quality parameters on the fingerprint recognition accuracy) has been the subject of some studies (Cappelli et al., 2008).

There are a number of live-scan sensing mechanisms (e.g., optical FTIR, capacitive, thermal, pressure-based, and ultrasound) that can be used to detect the ridges and valleys present on the fingertip. Figure 1.10 shows an off-line fingerprint image acquired with the ink technique, a latent fingerprint image, and some live-scan images acquired with different types of commercial live-scan devices. Optical fingerprint scanners have the longest history and are still used in applications where image quality specifications are stringent. Solid-state sensors gained popularity because of their compact size and the ease with which they can be embedded in consumer products such as laptop computers. This trend further accelerated in 2013 when Apple launched iPhone5S with a fingerprint sensor onboard, starting the era of mobile biometric authentication. In the years 2013–2018, small area capacitive sensors have been the default choice for mobile phone integration, even if their vulnerability was pointed out in a number of studies. Today, the small sensors

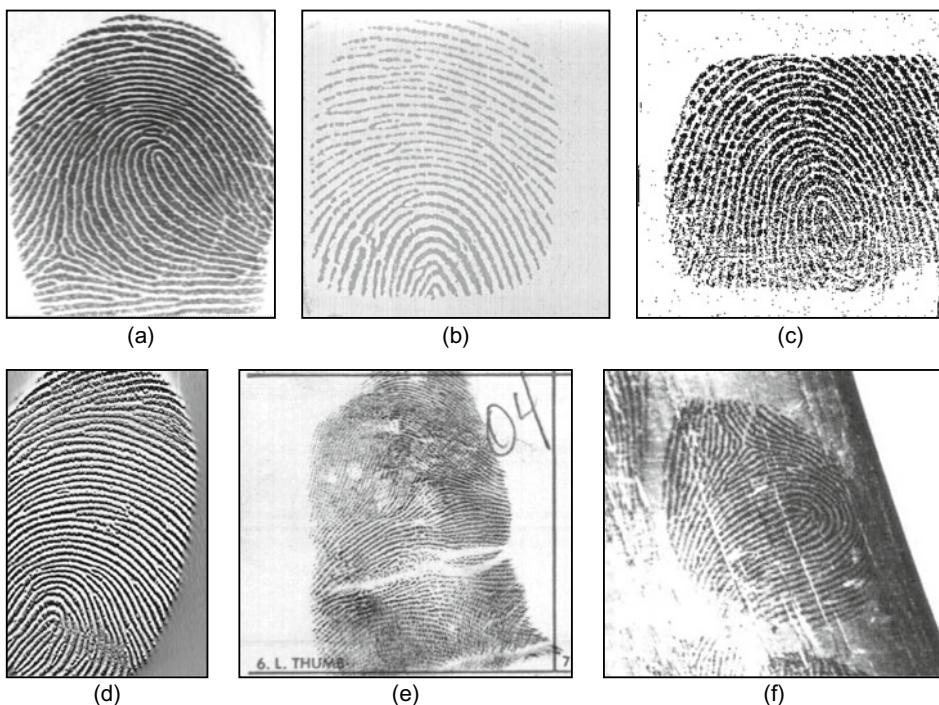


Fig. 1.10 Fingerprint images from **a** a live-scan FTIR-based optical scanner; **b** a live-scan capacitive scanner; **c** a live-scan piezoelectric scanner; **d** a live-scan thermal scanner; **e** an off-line inked impression; **f** a latent fingerprint

integrated into the home button of mobile phones are being replaced by optical or ultrasound in-display sensors, whose manufacturing cost can be reduced despite their large areas because of the CMOS → TFT transition. In fact, instead of using a silicon wafer, the TFT process deposits an array of tiny detectors on a (transparent) glass or plastic substrate. Since the production of flexible sensors is now feasible, payment cards with embedded fingerprint sensors are being introduced and could become one of the next massive-scale applications of fingerprint authentication.

Touchless fingerprint acquisition (both two or three-dimensional), introduced more than a decade ago, is receiving renewed interest for applications where multiple users have to interact with the same device, because of the lower risk of infection transmission (e.g., Covid-19).

Figure 1.11 shows some examples of fingerprint sensors embedded in a variety of computer peripherals and other devices.

Chapter 2 of this book discusses fingerprint sensing technologies, provides some characteristics of commercially available fingerprint scanners, and shows images acquired



Fig. 1.11 Fingerprint sensors can be embedded in a variety of devices and applications for user recognition

with a number of devices in different operating conditions (good-quality fingers, poor-quality fingers, and dry and wet fingers); storage of compressed fingerprint images is also introduced.

1.12 Fingerprint Representation and Feature Extraction

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

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

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

The fingerprint pattern, when analyzed at different scales, exhibits different types of features.

- *Level 1:* at the global level, the ridge-line flow delineates a pattern similar to one of those shown in Fig. 1.12. *Singular points*, called loop and delta (denoted as squares and triangles, respectively in Fig. 1.12), act as control points around which the ridge lines are “wrapped” (Levi & Sirovich, 1972). Singular points and coarse ridge-line shape are useful for fingerprint classification and indexing (see Chap. 5), but their distinctiveness is not sufficient for accurate matching. External fingerprint shape, orientation image, and frequency image also belong to the set of features that can be detected at the global level.

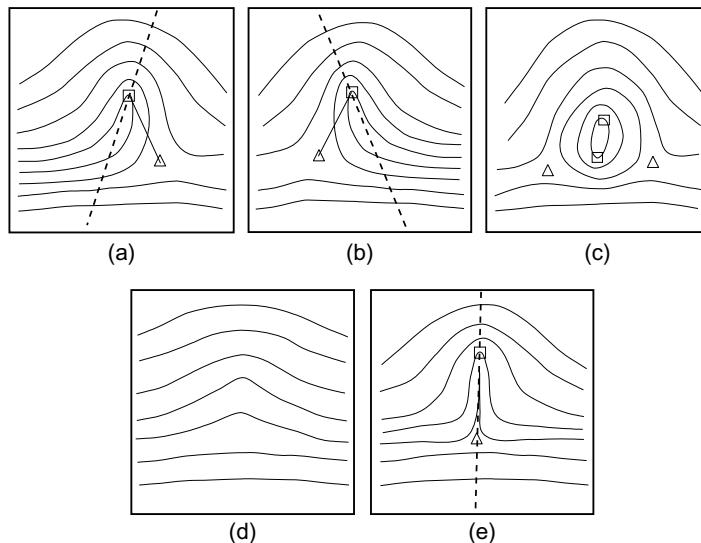
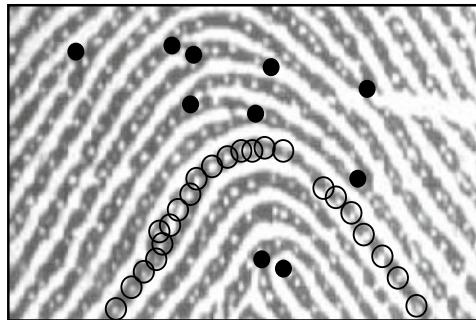


Fig. 1.12 Fingerprint patterns as they appear at a coarse level: **a** left loop, **b** right loop, **c** whorl, **d** arch, and **e** tented arch; squares denote loop-type singular points, and triangles delta-type singular points

Fig. 1.13 Minutiae (black-filled circles) in a portion of fingerprint image; sweat pores (empty circles) on a single ridge line



- *Level 2:* at the local level, a total of 150 different local ridge characteristics, called *minute details*, have been identified (Moenssens, 1971). These local ridge characteristics are not evenly distributed. Most of them depend heavily on the impression conditions and quality of fingerprints and are rarely observed in fingerprints. The two most prominent ridge characteristics, called *minutiae* (see Fig. 1.13) are *ridge endings* and *ridge bifurcations*. A ridge ending is defined as the ridge point where a ridge ends abruptly. A ridge bifurcation is defined as the ridge point where a ridge forks or diverges into branch ridges. Minutiae in fingerprints are generally stable and robust to fingerprint impression conditions. Although a minutiae-based representation is characterized by a high saliency, reliable automatic minutiae extraction can be problematic in extremely low-quality fingerprints devoid of any ridge structure.
- *Level 3:* at the very fine level, intra-ridge details can be detected. These include width, shape, curvature, edge contours of ridges as well as other permanent details such as dots and incipient ridges. One of the most important fine-level details is the finger *sweat pores* (see Fig. 1.13), whose positions and shapes are considered highly distinctive (although an opposite view is expressed in Monson et al., 2019). However, extracting very fine details including pores is feasible only in high-resolution (e.g., 1000 dpi) fingerprint images of good quality, and therefore this kind of representation is not practical for non-forensic applications.

In the last decade, machine learning approaches have been applied to fingerprint processing and feature extraction, initially based on dictionaries and bag-of-words approaches and more recently relying on successful deep learning models such as Convolutional Neural Network (CNN). This allowed the extraction of more reliable fingerprint representations, especially in very poor-quality fingerprints. On the other hand, successful image-processing algorithms, such as contextual filtering based on Gabor filters, remain the best choice when running on devices with limited computing resources.

Chapter 3 describes fingerprint anatomy and introduces the techniques available for processing fingerprint images and extracting salient features. Specific sections are dedicated to the definition and description of approaches for computing local ridge orientation, local ridge frequency, singular points, and minutiae. Particular emphasis is placed on fingerprint segmentation (i.e., isolation of fingerprint area [foreground or ridge-valley structure] from the background) and fingerprint image enhancement, which are very important intermediate steps in the extraction of salient features. Algorithms for estimating the quality of fingerprint images are also discussed.

1.13 Fingerprint Matching

Reliably matching fingerprints is a difficult problem, mainly due to the large variability in different impressions of the same finger (i.e., large intra-class variations). The main factors responsible for the intra-class variations are displacement, rotation, partial overlap, non-linear distortion, variable pressure, changing skin condition, noise, and feature extraction errors. Therefore, fingerprints from the same finger may sometimes look quite different, whereas fingerprints from different fingers may appear quite similar (see Fig. 1.14).

Human fingerprint examiners, in order to claim that two fingerprints are from the same finger, consider several factors: (i) global pattern configuration agreement, which means that two fingerprints must be of the same type, (ii) qualitative concordance, which requires that the corresponding minute details must be identical, (iii) quantitative factor, which specifies that at least a certain number (a minimum of 12 according to the forensic guidelines in the United States) of corresponding minute details must be found, and (iv) corresponding minute details, which must be identically interrelated. In practice, complex protocols have been defined for manual fingerprint matching, and a detailed flowchart is available to guide fingerprint examiners in manually performing fingerprint matching.

Automatic fingerprint matching does not necessarily follow the same guidelines. In fact, although automatic minutiae-based fingerprint matching is inspired by the manual procedure, a large number of approaches have been designed over the past 50 years, and many of them have been explicitly designed for automation. A (three-class) categorization of fingerprint matching approaches is as follows:

- *Correlation-based matching*: two fingerprint images are superimposed, and the correlation between corresponding pixels is computed for different alignments (e.g., various displacements and rotations).
- *Minutiae-based matching*: minutiae are extracted from the two fingerprints and stored as sets of points in the two-dimensional plane. Minutiae matching essentially consists of finding the alignment between the template and the input minutiae sets that results in the maximum number of minutiae pairings.

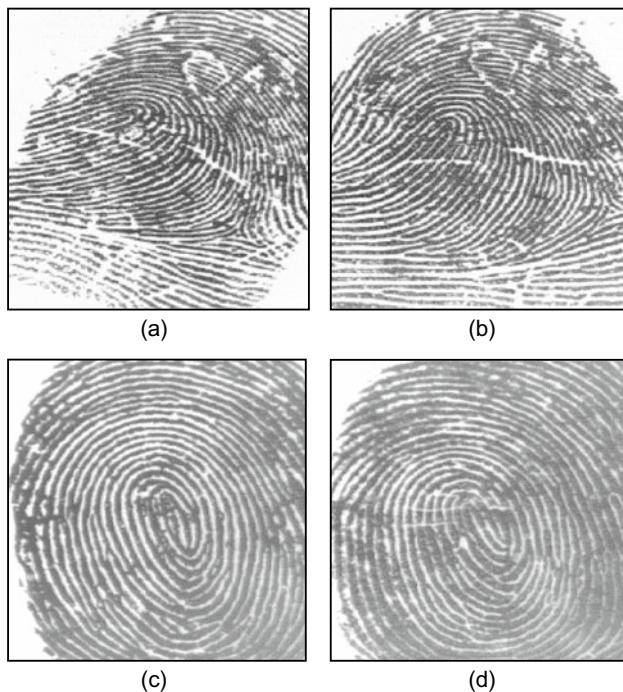


Fig. 1.14 Difficulty in fingerprint matching. Fingerprint images in **a** and **b** may look different to an untrained eye, but they are impressions of the same finger. Fingerprint images in **c** and **d** look similar to an untrained eye, but they are from different fingers

- *Feature-based matching:* minutiae extraction is difficult in extremely low-quality fingerprint images, whereas other features of the fingerprint ridge pattern (e.g., local orientation and frequency, ridge shape, and texture information) may be extracted more reliably than minutiae, even though their distinctiveness is generally lower. The approaches belonging to this family compare fingerprints in terms of features extracted from the ridge pattern, including handcrafted features such as SIFT, or representations learned by a convolutional neural network.

The accuracy of fingerprint recognition algorithms has steadily increased in the last 20 years. Although a direct comparison across the different competitions is not appropriate due to the use of databases of unequal difficulty, the accuracy of the best performing algorithms on FVC-onGoing is an order of magnitude better than the first competition held in 2000 (see Sect. 4.7.2):

- on FV-STD-1.0 database, which was collected under realistic operating conditions with a large-area sensor, $EER \cong 0.01\%$;
- on FV-HARD-1.0 database, which includes many challenging images, $EER \cong 0.2\%$.

However, there is still a need to continually develop more robust and accurate systems capable of properly processing and comparing poor-quality fingerprint images; this is particularly important when dealing with large-scale applications (e.g., civil registration or national ID programs), populations containing individuals who do manual labor with hands, or when small-area and low-cost sensors are employed, like in the current generation of smartphones.

Fingerprint recognition, like most biometric techniques, has been impacted by data-driven learning techniques. This has resulted in new effective methods for automated processing of latent fingerprints and learning robust fixed-length fingerprint representations. However, top-down minutiae-based “geometric” matching still remains the best-performing approach for most use cases of fingerprint recognition. This shows that tiny ridge details, introduced for fingerprint comparison by Sir Francis Galton more than a century ago, are still competitive with the powerful representations learned by huge neural networks trained on large sets of fingerprints.

Chapter 4 is dedicated to the fingerprint matching problem. The matching problem is formally presented, the above three classes of matching techniques are discussed, and the related literature is surveyed in detail. Particular emphasis is given to minutiae matching and both global and local minutiae matching algorithms are introduced. The best accuracy versus efficiency trade-off can be achieved by combining local and global minutiae matching into a two-phase approach where an initial local structure matching is followed by a consolidation step. This technique is extensively discussed in Sect. 4.4. A separate section is then dedicated to non-linear distortion affecting fingerprint impressions and to the design of distortion-tolerant matchers.

1.14 Fingerprint Classification and Indexing

Large volumes of fingerprints are collected and stored every day in a wide range of applications, particularly in forensics and government applications, e.g., background check of employees. Automatic identification based on fingerprints requires matching the input (query or test) fingerprint against a large number of templates stored in a database. To reduce the search time and computational complexity, several pre-selection techniques have been proposed and adopted that can be categorized as (1) exclusive classification and (2) indexing (also referred to as continuous classification). Fingerprint exclusive classification is a technique used to assign a fingerprint to one of the several pre-specified types (see Fig. 1.12). An input fingerprint is first classified into one of the pre-specified types, and then it is compared to a subset of the database corresponding to that fingerprint

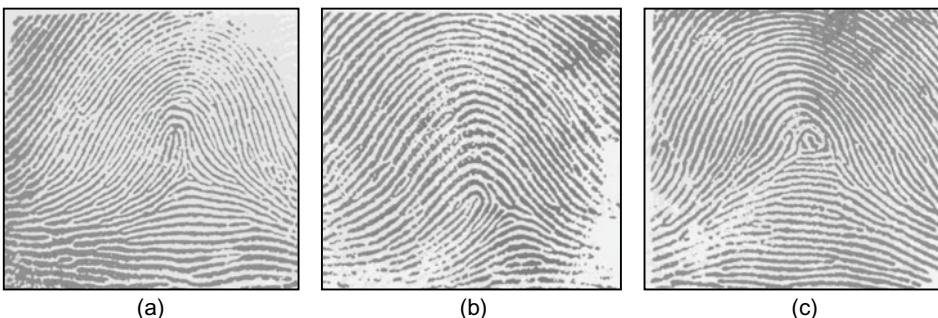


Fig. 1.15 Examples of fingerprints that are difficult to classify; **a** a tented arch; **b** a loop; **c** a whorl; it appears that all the fingerprints shown here should be in the loop category

type (i.e., the pre-selection algorithm selects a bin to reduce the number of templates to be matched). A well-known classification of fingerprints was proposed by Henry (see Lee & Gaenssen, 2012) that consists of five major classes: whorl, left loop, right loop, arch, and tented arch. If the fingerprint database is binned into these five classes, and a fingerprint classifier outputs two classes (primary and secondary) with extremely high accuracy, then the identification system will only need to search two of the five bins, thus the pre-selection algorithm will decrease (in principle) the search space by 2.5-fold. Unfortunately, the distribution of fingerprints even into these five categories is not uniform, and there are many “ambiguous” fingerprints (see Fig. 1.15) that cannot be accurately classified even by human experts. About 17% of the 4000 images in the NIST Special Database 4 (Watson & Wilson, 1992) have two different ground truth labels! Therefore, in practice, fingerprint classification is not immune to errors and does not offer much selectivity for fingerprint searching in large databases. For these reasons, exclusive classification techniques, as methods of pre-selection in identification systems, today have only a historical value, while retaining their validity in other contexts, i.e., for anthropological studies.

To overcome the above problem, fingerprint indexing techniques can be used. The key idea, regardless of the numerous variants and implementations, is to select an appropriate representation and an efficient data structure suitable for making the search operation efficient by reducing the number of candidates. Indexing is coupled with a retrieval strategy aimed to create a list of candidates to be submitted to the matching stage. Hashing techniques based on local descriptors (e.g., minutiae triplets or MCC) have been the primary approach for several years, but recently they have been supplanted by fixed-length global representation extracted by deep learning-based approaches, especially for rolled fingerprints.

Chapter 5 covers fingerprint classification and indexing techniques and the related pre-selection algorithms (also known as retrieval algorithms). Metrics, datasets, and performance evaluation for classification and indexing are also introduced.

1.15 Latent Fingerprint Recognition

Latent fingerprint recognition is a challenging problem because a latent fingerprint usually contains much less information than fingerprints obtained by live-scan devices and the inking method. The fact that latent fingerprint recognition usually works in the identification mode (1:N search) for a large database makes the latent search even more challenging. To improve the recognition accuracy of latent fingerprints, human experts were generally required in feature extraction and verification stages in the past. However, there are some drawbacks in manual latent fingerprint recognition, including slow speed, high labor cost, and low compatibility with the AFIS.

In the past decade, a number of research papers have been published on latent fingerprint recognition, driven by the need for higher accuracy and automation. Substantial progress has been made in this field. According to the fact sheet of the FBI's NGI system, in February 2021, no manual feature marking was required for about 20% of latent search.

Chapter 6 is devoted to latent fingerprints. The manual latent fingerprint recognition method and its limitations are first introduced. Feature extraction and matching methods which are specifically developed for latent fingerprints are then presented in detail.

1.16 Synthetic Fingerprints

Performance evaluation of fingerprint recognition systems is data-dependent. Therefore, the acquisition conditions, database size, and confidence intervals must be specified when reporting the matching results. Typically, to obtain tight confidence intervals at very low error rates, large databases of representative fingerprint images are required. Moreover, once a fingerprint database has been used for testing and optimizing a system, successive testing cycles require new sequestered databases, previously unseen by the system.

The collection of large fingerprint databases is expensive both in terms of time and money. There are also problems due to data collection errors and privacy legislation protecting the use and sharing of personal data. Several fingerprint datasets available to the scientific community have been recently withdrawn due to privacy regulations. In several contexts, a synthetic generation of realistic fingerprint images may alleviate these problems.

The proposed fingerprint synthesis approaches can be grouped into two main categories:

- Methods that first generate a master fingerprint and then derive synthetic impressions from it. A master fingerprint is a noise-free pattern that encodes the unique and immutable characteristics of a “synthetic finger”, independently of the variations (displacement, rotation, pressure, skin condition, distortion, noise, etc.) that make the

successive acquisitions different from each other. The SFinGe approach (Cappelli et al., 2000a, 2002) is the best known technique in this category.

- Methods that directly generate each synthetic fingerprint starting from a set of input parameters. Deep generative models (e.g., GAN) have been recently used to directly generate very realistic fingerprint samples.

The use of synthetic fingerprints is not only limited to the problem of performance evaluation; in fact, their utility has been demonstrated for (i) data augmentation and boosting for the training of machine learning models, (ii) testing the robustness of fingerprint verification systems against attacks, and (iii) semantic conformance and interoperability studies.

Chapter 7 introduces synthetic fingerprint generation and presents in detail the most effective approaches. Due attention is paid to the validation of synthetic generators, also referring to recent large-scale experiments. The SFinGe software tool (demo version), included in the ESM that accompanies this book, can be used to create a synthetic fingerprint step by step, observing the effects of various parameter values on the resulting fingerprint image (Fig. 1.16).

1.17 Biometric Fusion

How can the performance of a fingerprint recognition system be improved? There comes a stage in the development of any biometric recognition system where it becomes increasingly difficult to achieve significantly better performance from a given biometric identifier or a given recognition method. There often is a need to explore other sources and methods for improvement in recognition accuracy. The *fusion* approach to improve performance can take any number of different forms. One may fuse multiple biometric traits or multiple instances of the same biometric trait or even complimentary feature extraction and matching algorithms for the same instance of a biometric trait. Performance gains have been reported from various such fusion approaches.

Fusing multiple biometric identifiers can also alleviate several practical problems in biometrics-based personal recognition. For instance, although a biometric identifier is supposed to be *universal* (each person in the target population should possess it), in practice, no biometric identifier is truly universal. Similarly, it is not always possible to sense all the biometric identifiers (non-zero failure to acquire rate). Biometric identifiers are also prone to failure to enroll. That is, there are some users in the target population whose biometric identifiers are not easily quantifiable by the given biometric sensor. Consequently, the recognition system cannot handle these users based on that particular biometric identifier. Fusion of face, fingerprints, and irises was found to be very important for India's Aadhaar program during the large-scale de-duplication stage. In highly secure systems, reinforcement of evidence from multiple biometric identifiers offers increasingly irrefutable proof



Fig. 1.16 Synthetic fingerprint images generated with the software tool SFinGe

of the identity of the authorized person. It is also extremely difficult for an intruder to fake several different biometric traits of a genuine user in order to circumvent the system. The assumptions of universality, collectability, acceptability, and integrity are more realistically accommodated when person recognition is based on information from several biometric identifiers. Due to the abovementioned advantages, multibiometric systems are popular in various person recognition applications, ranging from time and attendance, smartphone unlock to border crossing, and national ID cards.

Multiple modalities of biometrics can be combined at the sensor, feature, matcher score, or matcher decision levels (Brooks & Iyengar, 1997). The integration at the sensor or feature level assumes a strong interaction among the input measurements and such integration schemes are referred to as *tightly coupled integrations* (Clark & Yuille, 1990). The *loosely coupled integrations*, on the other hand, assume very little or no interaction among the inputs (e.g., face and finger) and the integration occurs at the output of relatively autonomous agents, each agent independently assessing the input from its own perspective.

The focus of most biometric fusion research has been on loosely coupled integration. The loosely coupled integration is not only simpler to implement, but it is also more feasible in commonly confronted integration scenarios. A typical scenario for integration is two biometric systems (often proprietary) independently acquiring inputs and making an autonomous assessment of the “match” based on their respective identifiers; although the decisions or matching scores of individual biometric systems are available for integration, the features used by one biometric system are usually not accessible to the other biometric system. Decision-level and matcher score-level integration can deliver at least as good or better performance than any single constituent biometric trait (Hong et al., 1999).

Tightly coupled integration is much harder. International Standards Organization (ISO) has introduced standards on interoperability that have led to common formats of finger-print representations for easy exchange of data among vendors. A limitation of these approaches is that these schemes force vendors to use the least common denomination of the representation as a basis of data sharing (e.g., minutiae) and consequently, there is significant degradation in performance when one vendor is using the features extracted by another vendor (Grother et al., 2006).

Considering that biometric fusion is a relatively mature and widely implemented technology, we do not have a separate chapter on biometric fusion. Interested readers who need more details may refer to Chap. 7 of the second edition of this book, the book on multibiometrics (Ross et al., 2006), and survey papers (Dinca & Hancke, 2017; Singh et al., 2019).

1.18 System Integration and Administration Issues

The major issues in using a fingerprint recognition system in an application include defining the system working mode (verification or identification), choosing hardware (e.g., fingerprint scanner) and software (e.g., feature extraction and matching algorithms) components and making them work together, dealing with exceptions and poor-quality fingerprint images, and defining effective administration and optimization policy. These issues are often left to the application administrator. There are lots of core fingerprint recognition components (scanners and algorithms) available from a variety of vendors. With the advances in interoperability standards, different applications may choose to mix and match such components to achieve the objectives of the application (e.g., the performance versus cost).

As mentioned in Sect. 1.3, a fingerprint-based system may operate either in verification or identification mode. As a rule of thumb, when the number of users is large (>1000), it is recommended that the system designer choose the verification mode unless identification is strictly necessary.¹ This is because the verification system scales very well, i.e., the verification response time, accuracy, and resource consumption (such as system memory and processing power) do not depend on the size of the database. On the other hand, in identification, the response time decreases as the database size increases. Further, the accuracy degrades, and more system resources (such as memory and processing power) are required as the database size increases. In case neither binning nor clever indexing/retrieval mechanisms are available, an identification system needs to explore the

¹ Applications that assume a negative claim of identity cannot work in verification mode: in fact, the system has to search the entire archive to prove that the query feature set does not have a match in the enrollment database. Sometimes, even in applications that assume positive claim of identity, the system must necessarily work in identification mode, due to the practical difficulty of using an input device to enter a PIN.

entire template database to establish an identity. Even if indexing is used, it is doubtful that a one-finger matching can reach the desirable efficiency and accuracy on a large database. As a result, the use of multiple fingers is recommended in medium- (tens of thousand) to large-scale (tens of million) identification applications.

If the system designer or integrator is also the developer of the feature extraction and matching (and eventually indexing) algorithms, then she certainly has the necessary knowledge to combine all the modules and to select the optimal fingerprint scanner and computing platform. In the biometric field, developers and system integrators are not always the producers of fingerprint scanners and core matching algorithms, and therefore, care must be taken when choosing acquisition scanners and matching algorithms (hardware and software components). The system designer should take into account several factors:

- *Proven technology*: have the hardware and software components been tested by third parties? Are the test results available? Is the vendor available to demonstrate that the claimed performance (accuracy and response time estimates) is true?
- *System interoperability and standards*: is the system compliant with emerging standards? Is the software compliant with all the platforms and operating systems of interest?
- *Cost versus performance trade-off*: the optimal point in the cost versus performance trade-off strongly depends on the application requirements. The cheapest solution is not necessarily the best choice; biometrics is not infallible, and the success of an application often depends on how much of the customer's expectation is met.
- *Support*: available documentation, examples, etc.

Vendors may supply a Software Development Kit (SDK) in the form of libraries for one or more operating systems. These libraries typically include a series of primitives that allow different tasks to be performed (e.g., fingerprint acquisition, feature extraction, template creation, matching, and template storage). The system designer is usually in charge of developing specific routines for

- Implementing the enrollment stages.
- Storing and retrieving templates and user information in/from a centralized/ distributed template storage (database).
- Defining the user search order in an identification application. For example, the template of the users most frequently accessing the system may be matched before those of infrequent users.
- Defining policies and administration modules to let the system administration define and control the system behavior. This includes setting the system security options (system threshold, number of trials, alarms, etc.) and logging information about access attempts.

An important point when designing a fingerprint recognition system is to decide from the beginning how to deal with users whose fingerprint quality is extremely poor. Although the percentage of users with “unusable” fingerprints is minuscule, it cannot be ignored, especially in large-scale applications such as civil registration. There are several options to deal with such a problem:

- In the enrollment stage, choose the best quality finger and eventually enroll more fingers or more instances of the same finger.
- Define user-dependent system thresholds. In particular, the system threshold may be relaxed by the system administrator for fingers that are hard to match (to reduce false non-match rate); for other users the threshold is maintained at the default level. Although this has serious security implications, it may be preferable to decreasing the system operating point for all the users, because an impostor who intends to “imitate” an enrolled user usually is not aware of which users have poor-quality fingers.
- Use additional biometric trait(s) (multimodal biometric system).
- Use non-biometric information. For example, using a computer-generated difficult password (also called OTP or One Time Password) could be an acceptable authentication alternative for a limited number of users.

System administration and optimization are also very important issues. An administrator (or proper documentation) should briefly instruct users the first time they use the system and, in particular, make them familiar with the use of fingerprint scanners. An attended enrollment is often preferable to check the quality of input, select the best finger, and eventually relax the user-dependent threshold (if this option is available). The administrator is also in charge of setting the global security threshold, controlling the state of the acquisition devices (the live-scan scanners sometimes become dirty over time and therefore, the quality of the input images tends to deteriorate), and monitoring access attempts. In particular, in case some users find it difficult to access the system, the administrator should understand the underlying reasons: a new enrollment could solve the problem in the case of some physical variations in the finger (e.g., a recent injury or scratch); retraining users on how to properly interact with the scanner could be sufficient in several other cases. Monitoring the system log could also be very useful to discover if the system is being subjected to attacks by fraudulent users.

1.19 Securing Fingerprint Systems

Fingerprint recognition systems are security systems and as such they are not foolproof. Despite numerous advantages, fingerprint systems are vulnerable to security breaches and attacks. The system vulnerability depends on the threat model of an application which

will use the fingerprint recognition system. The typical threats in a fingerprint recognition system are as follows:

- *Denial of service*: an adversary can damage the system to cause a denial of service to all the system users.
- *Circumvention or intrusion*: an unauthorized user can illegitimately gain access into the system (including, by colluding with the system administrator or by coercing a legitimate authorized user).
- *Function creep*: a wrongful acquisition or use of fingerprint data for a purpose other than intended.
- *Repudiation*: a legitimate user may deny having accessed the system.

The first step in analyzing the security of fingerprint systems is defining the threat model. Bolle et al. (2002) and Ratha et al. (1999, 2001, 2003) describe an “attack point” model to describe vulnerabilities in biometric systems. Cukic and Bartlow (2005) adopt an “attack tree” model, while Jain et al. (2006) adopt a “fishbone” model. In Chap. 9, we adopt a slightly different viewpoint in looking at the types of fingerprint system failures, how the failures can be triggered, and discuss some of the techniques that have been developed as countermeasures. It is expected that multiple techniques will be used in building a practical system depending on the application’s threat model.

In Chap. 9, we will focus primarily on the intrusion threat. For a successful intrusion, a hacker needs to first obtain fingerprint data and then inject it into the authentication system. There are different ways by which a hacker can obtain fingerprint data and different ways by which the hacker can inject the data into the system. We will present countermeasures for each.

The most critical user interface of a biometric system is the interaction of the user with the scanner. The vulnerability of fingerprint systems to presentation attacks and numerous media reports of successful presentation attacks have raised concerns about the security of fingerprint systems. This is especially true in the case of fingerprint systems operating in an unsupervised scenario (such as fingerprint systems in a smartphone). Hence, the topic of presentation attack detection (PAD) has received enormous attention over the past two decades. A number of hardware-based and software-based PAD approaches have been proposed. Of particular interest among the various biometric circumvention measures is that of checking whether the source of the input signal is a live and original (i.e., not dead or fake). The premise of a liveness test is that if the finger is live and original, the impression made by it represents the person to whom the finger belongs. One of the approaches of detecting liveness is to measure one or more vital signs (e.g., pulse and temperature) of life in the object being imaged. High-resolution fingerprint scanning can reveal the characteristic sweat pore structure of the skin (Roddy & Stosz, 1997) that is difficult to replicate in an artificial finger. The skin tone of a live finger turns white or yellow when pressed against a glass platen. This effect can be exploited for detecting a live finger. The

blood flow in a live finger and its pulsation can be detected by a careful measurement of light reflected or transmitted through the finger. Difference in action potentials across two specific points on a live fingerprint muscle can also be used to distinguish it from a dead finger. The electrical properties of a live finger are ascertained rather effortlessly in some solid-state fingerprint scanners. Measuring the complex impedance of the finger can be a useful attribute to distinguish a live finger from its lifeless counterpart. A live finger generates sweat and this sweating process can be monitored to determine liveness. Any combination of pulse rate, electrocardiographic signals, spectral characteristics of human tissue, percentage of oxygenation of blood, blood flow, hematocrit, biochemical assays of tissue, electrical plethysmography, transpiration of gases, electrical property of skin, blood pressure, and differential blood volumes can be used to detect a live finger.

A good strategy to secure the modules, communication channels, and template storage of a fingerprint recognition system from the hackers is to design the system such that it is a closed system. A closed system in this context means that the modules of the system trust each other but no one else. We discuss techniques to build a closed system in Chap. 9. In particular, the popular match-on-card, system-on-device, system-on-card, and system-on-a-chip architectures are discussed.

Protection of biometric template has received significant attention from the research community. In a knowledge- or token-based system, if the knowledge or token is compromised, it can be changed, but if a biometric template is stolen, it can be used to circumvent the fingerprint system repeatedly as the user cannot change her biometric trait. Template protection techniques are discussed in Chap. 9.

1.20 Privacy Issues

Privacy is the ability to lead one's own life free from intrusions, to remain anonymous, and to control access to one's own personal information. Since privacy deals with personal information, there needs to be an enforceable, objective definition of a person's identity. As the magnitude of identity fraud increases and as we are increasingly being asked to prove our identity to strangers in remote locations, there appears to be a tendency to lower the standards of suspecting the validity of claimed identity for authorizing transactions. Biometrics such as fingerprints will increasingly come into play for positively recognizing people because of the limitations of the conventional technologies (e.g., knowledge-based or token-based). For instance, the US legislation requires the use of strong recognition schemes (such as biometric identifiers) for controlling access to sensitive medical records to authorized personnel. Some applications have envisaged using biometrics for anonymous access. For instance, these applications index sensitive individual information without explicitly specifying a name and the access mechanisms (e.g., allow access to medical records if the person's left index fingerprint matches the fingerprint associated with this record). Furthermore, by requiring automated access mechanisms through a

secure biometric system, it is hoped that all the accesses to the privileged information can be tracked, thereby increasing the accountability of transactions within the information systems. Thus, it is clear that fingerprints will be useful for enhancing the integrity of systems holding personal information.

In spite of the security benefits offered by biometric recognition, there are objections to biometric recognition based on the following arguments. Methods of automatic recognition of individuals based on biometrics may be perceived as demeaning. Religious objections interpret biometric recognition as “the mark of the beast” by citing somewhat dubious biblical references.² Some potential users have raised concerns about the hygiene of biometric scanners requiring contact. Given that we routinely touch many objects (e.g., money) touched by strangers, this objection may be viewed as a frivolous excuse. However, what was a minor concern become a more serious risk with the diffusion of the Covid-19 pandemic in 2020, and today touchless fingerprint acquisition is being considered for border control gates.

There may be negative connotations associated with some biometrics (fingerprint, face, and DNA) due to their prevalent use in criminal investigations.

There are some other stronger criticisms being leveled against the unintended but potentially harmful (to the user) capabilities of biometric identifiers.

- *Unintended functional scope:* because biometric identifiers are based on human anatomy, additional (possibly statistical) personal information may be gleaned from the scanned biometric measurements. For instance, it is known that malformed fingers may be statistically correlated with certain genetic disorders (Babler, 1991; Penrose, 1965; Mulvhill, 1969). With advancements in human genome research, the fear of inferring personal information from biological measurements may be imminent. Such derived medical information may become a basis for systematic discrimination against the perceived “risky” or “undesirable” sections of the population.
- *Unintended application scope:* persons legally maintaining multiple identities (say, for safety reasons under a witness protection program) can be detected based on their fingerprints. By acquiring biometrics identifiers (either covertly or overtly), one has the capacity to track a person of interest. It has been argued that automatically gathering individual information based on biometric identifiers accrues unfair advantage to people in power and reduces the sovereignty of private citizens. In the case of fingerprints, presently there is no technology to automatically capture fingerprints covertly to facilitate effortless tracking.³ Yet, persons who desire to remain anonymous in any

² “He also forced everyone, small and great, rich and poor, free and slave, to receive a mark on his right hand or on his forehead, so that no one could buy or sell unless he had the mark, which is the name of the beast or the number of his name.” (Revelation 13:16–17).

³ Although there is touchless (direct) fingerprint scanning technology available, it is still necessary for the subject to be in the very close proximity of the scanner. There is presently no technology capable of video snooping of fingerprints.

particular situation may be denied their privacy as their fingerprint templates from different systems may be linked together through fingerprint matching.

The possible abuse of biometric information (or their derivatives) and related accountability procedures can be addressed through government legislation (e.g., EU General Data Protection Regulation (GDPR)⁴), assurance of self-regulation by the biometric industry (e.g., self-regulation policies of the International Biometrics Industry Association [IBIA]), and autonomous enforcement by independent regulatory organizations (e.g., a Central Biometric Authority). Until such consensus is reached, there may be reluctance by some users to provide their biometrics measurements. As a result, applications delivering recognition capability in a highly decentralized fashion are likely to be favored.

In verification applications, one way to decentralize a biometric system is by storing the biometric information in a decentralized (encrypted) database over which the individual has complete control. For instance, one could store the fingerprint template in a tamper-resistant smartcard that is issued to the user. The input fingerprint feature set can be directly compared with the template on the smartcard and the decision delivered (possibly in encrypted form) to the application. Thus, the template is never released from the secure storage of the smartcard. Such a smartcard-based system permits all the advantages of biometric recognition without many of the stipulated privacy problems associated with biometrics. Commercial products already exist that implement system-on-a-chip, where the fingerprint scanner is directly connected to the physically secure chip that stores the templates as well as conducts the entire fingerprint processing, from fingerprint acquisition to feature extraction to matching, on the secure chip. Analogously, in modern mobile phones, fingerprints are stored in a Trusted Execution Environment (TEE),⁵ where the information is encrypted and kept in a secure area of the main processor that cannot be accessed by the user or the applications. The operating systems can ask the TEE to verify a user's identity using biometrics, but it cannot extract (or export) the biometric information.

See Chap. 9 for a discussion on match-on-card and system-on-a-chip solutions. Chapter 9 also discusses template protection techniques, which have privacy protection and privacy-enhancing properties. These methods are currently under intense research and development.

⁴ <https://gdpr.eu>.

⁵ https://en.wikipedia.org/wiki/Trusted_execution_environment.

1.21 Summary and Future Prospects

Fingerprint recognition has come a long way since its inception more than 100 years ago. The first primitive live-scan scanners designed by Cornell Aeronautical Lab/North American Aviation, Inc. in the 1970s were unwieldy beasts with many problems as compared to the sleek, inexpensive, and compact scanners available today. Over the past few decades, research and active use of fingerprint matching and indexing have also advanced our understanding of the power as well as limitations of fingerprint recognition. A steady increase in processor power and memory capacity at lower prices, cheap and compact fingerprint scanners, and growing demand for security have led to the viability of fingerprint matching for routine person recognition tasks.

After five decades of intensive research, there is a popular misconception that automatic fingerprint recognition is a fully solved problem. There are still a number of challenges in designing completely automatic and reliable fingerprint recognition algorithms, especially for poor-quality fingerprint images as well as latent fingerprints. Although state-of-the-art automated fingerprint systems have impressive performance, they still cannot match the performance of a fingerprint expert in handling poor-quality and latent fingerprints. On the other hand, automated fingerprint recognition systems offer a reliable, rapid, consistent, and cost-effective solution in a number of traditional and emerging applications that require person recognition. Some of the difficult problems in fingerprint recognition will entail solving not only the core pattern recognition challenges but also confronting some challenging system engineering issues related to security and privacy.

In many computer vision applications (e.g., object recognition), the state of the art is nowadays achieved with deep learning techniques. The same is true for many biometric traits (e.g., face recognition). Fingerprint recognition is still heavily relying on hand-crafted features and classical image processing algorithms, and the most effective machine learning approaches to fingerprint recognition exploit domain knowledge to improve their accuracy. New ideas and novel machine learning approaches could change the scenario in the future and extend the machine learning “supremacy” also to this field.

Only a few years back, it seemed as though interest in fingerprint matching research was waning. As mentioned earlier, due to a continuing increase in identity fraud, there is a growing need for positive person recognition. Lower fingerprint scanner prices, inexpensive computing power, and our (relatively better) understanding of individual information in fingerprints (compared to other biometric traits) have attracted a lot of commercial interest in the fingerprint-based recognition system. Fingerprint-based mobile authentication, pioneered by Apple in 2013, is nowadays pervasive, and emerging large area sensors embedded in the smartphone screen could pave the way for new authentication options such as multi-finger recognition for high-security applications and finger-specific actions.

1.22 Image Processing, Pattern Recognition, and Machine Learning Background

Some background in image processing, pattern recognition, and machine learning techniques is necessary to fully understand a majority of the chapters in this book (especially Chaps. 3–7). We recommend the following books and periodicals to readers who do not have this background.

1.22.1 Image Processing Books

- Richard Szeliski, *Computer Vision: Algorithms and Applications*, 2nd ed., <https://szeliski.org/Book>, 2021
- R.C. Gonzalez and R.E. Woods, *Digital Image Processing* (4th edition), Pearson, 2018.
- J. Bigun, *Vision with Direction: A Systematic Introduction to Image Processing and Computer Vision*, Springer, NY, 2006.
- D.A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach* (2nd edition), Pearson, 2011.
- L.G. Shapiro and G. Stockman, *Computer Vision*, Prentice-Hall, Englewood Cliffs, NJ, 2001.
- R. Bracewell, *The Fourier Transform and Its Applications* (3rd edition), McGraw-Hill, New York, 1999.
- S. Mallat, *A Wavelet Tour of Signal Processing*, Academic, New York, 1997.
- J. Parker, *Algorithms for Image Processing and Computer Vision*, John Wiley, New York, 1996.
- A.K. Jain, *Fundamentals of Digital Image Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1988.

1.22.2 Pattern Recognition and Machine Learning Books

- Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and Tensorflow: Concepts, Tools, and Techniques to Build Intelligent Systems*, O'reilly & Associates, 2019.
- Pang-Ning Tan, Michael Steinbach, Anuj Karpatne and Vipin Kumar, *Introduction to Data Mining* (2nd edition), Pearson, 2018.
- Charu Aggarwal, *Neural Networks and Deep Learning: A Textbook*, Springer, 2018.
- I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*, MIT Press, 2016.
- S. Haykin, *Neural Networks and Learning Machines*, Prentice Hall, 2008.

- R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification* (2nd edition), Wiley-Interscience, New York, 2000.
- C.M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, 1995.
- A.K. Jain and R.C. Dubes, *Algorithms for Clustering Data*, Prentice-Hall, Englewood Cliffs, NJ, 1988.

1.22.3 Journals

- *IEEE Transactions on Pattern Analysis and Machine Intelligence*
- *IEEE Transactions on Information Forensics and Security*
- *IEEE Transactions on Biometrics, Behavior, and Identity Science*
- *IEEE Transactions on Image Processing*
- *Pattern Recognition*
- *Pattern Recognition Letters*
- *IET Biometrics*.

References

- Babler, W. J. (1991). Embryologic development of epidermal ridges and their configuration. *Birth Defects Original Article Series*, 27(2), 95–112.
- Bazen, A. M., & Veldhuis, R. N. J. (2004). Likelihood-ratio-based biometric verification. *IEEE Transaction on Circuits and Systems for Video Technology*, 14(1), 86–94.
- Bolle, R. M., Connell, J. H., & Ratha, N. K. (2002). Biometric perils and patches. *Pattern Recognition*, 35(12), 2727–2738.
- Bolle, R. M., Ratha, N. K., & Pankanti, S. (1999). Evaluating authentication systems using bootstrap confidence intervals. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 9–13).
- Bolle, R. M., Ratha, N. K., & Pankanti, S. (2004). Error analysis of pattern recognition systems—The subsets bootstrap. *Computer Vision and Image Understanding*, 93(1), 1–33.
- Bonneau, V., Probst, L., & Lefebvre, V. (2018). Biometrics technologies: A key enabler for future digital services. Retrieved July, 2021, from <https://ati.ec.europa.eu/reports/technology-watch/biometrics-technologies-key-enabler-future-digital-services>.
- Brooks, R. R., & Iyengar, S. S. (1997). *Multi-sensor fusion: Fundamentals and applications with software*. Prentice-Hall.
- Cappelli, R., Ferrara, M., & Maltoni, D. (2008). On the operational quality of fingerprint scanners. *IEEE Transactions on Information Forensics and Security*, 3(2), 192–202.
- Cappelli, R., Maio, D., & Maltoni, D. (2000a). Synthetic fingerprint-image generation. In *Proceedings of International Conference on Pattern Recognition* (15th ed., Vol. 3, pp. 475–478).

- Cappelli, R., Maio, D., & Maltoni, D. (2000b). Indexing fingerprint databases for efficient 1:N matching. In *Proceedings of International Conference on Control, Automation, Robotics and Vision* (6th ed.).
- Cappelli, R., Maio, D., & Maltoni, D. (2002). Synthetic fingerprint-database generation. In *Proceedings of International Conference on Pattern Recognition* (16th ed.).
- Cappelli, R., Maio, D., Maltoni, D., Wayman, J. L., & Jain, A. K. (2006). Performance evaluation of fingerprint verification systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1), 3–18.
- Clark, J., & Yuille, A. (1990). *Data fusion for sensory information processing systems*. Kluwer.
- Cole, S. A. (2001). What counts for identity? *Fingerprint Whorld*, 27(103), 7–35.
- Cukic, B., & Bartlow, N. (2005, September). Biometric system threats and countermeasures: A risk based approach. In *Proceedings of Biometric Consortium Conference*, Crystal City, VA, USA.
- Cummins, H., & Midlo, C. (1943). *Fingerprints, palms and soles*. Dover.
- Cummins, H., & Midlo, C. (1961). *Fingerprints, palms and soles: An introduction to dermatoglyphics*. Dover.
- Daugman, J. (1999). Recognizing persons by their iris patterns. In A. K. Jain, R. Bolle, & S. Pankanti (Eds.), *Biometrics: Personal identification in a networked society*. Kluwer.
- Dinca, L. M., & Hancke, G. P. (2017). The fall of one, the rise of many: A survey on multi-biometric fusion methods. *IEEE Access*, 5, 6247–6289.
- Doddington, G., Ligget, W., Martin, A., Przybocki, M., & Reynolds, D. (1998). Sheep, goats, lambs, wolves: An analysis of individual differences in speaker recognition performance. In *Proceedings of International Conference on Speech and Language Processing*.
- Engelsma, J. J., Cao, K., Jain, A. K. (2021). Learning a fixed-length fingerprint representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(6), 1981–1997.
- FBI. (2021, April). FBI biometric services section. Next Generation Identification (NGI) System Fact Sheet. Retrieved June, 2021, from <https://www.fbi.gov/file-repository/ngi-monthly-fact-sheet>.
- FBI—CJIS Division. (2006). Image quality specifications for single finger capture devices. Retrieved March, 2022, from <https://fbibiospecs.fbi.gov/file-repository/pivspec.pdf/view>.
- FBI—CJIS Division. (2017). *Electronic Biometric Transmission Specification (EBTS)*. Int. Report: NGI-DOC-01862-1.1 (V10.0.8). Retrieved July, 2021, from https://www.fbibiospecs.cjis.gov/Document/Get?fileName=Master%20EBTS%20v10.0.8%2009302017_Final.pdf.
- Federal Bureau of Investigation. (1984). *The science of fingerprints: Classification and uses*. U.S. Government Publication.
- Federal Bureau of Investigation. (1991). *The FBI fingerprint identification automation program: Issues and options*. U.S. Government Publication/Congress of the U.S., Office of Technology Assessment.
- Galton, F. (1892). *Finger prints*. Macmillan.
- Golfarelli, M., Maio, D., & Maltoni, D. (1997). On the error-reject tradeoff in biometric verification systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7), 786–796.
- Grother, P., McCabe, M., Watson, C., Indovina, M., Salamon, W., Flanagan, P., Tabassi, E., Newton, E., & Wilson, C. (2006, March). *Performance and interoperability of the INCITS 378 fingerprint template*. NIST Research Report: NISTIR 7296.
- Hong, L., Jain, A. K., & Pankanti, S. (1999). Can multibiometrics improve performance? In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 59–64).
- ISO/IEC 19795-2. (2007). ISO/IEC, ISO/IEC 19795-2:2007—Biometric performance testing and reporting—Part 2: Testing methodologies for technology and scenario evaluation. ISO/IEC Standard.

- ISO/IEC 19795-4. (2008). ISO/IEC, ISO/IEC 19795-4:2008—Biometric performance testing and reporting—Performance and Interoperability Testing of Interchange Formats. ISO/IEC Standard.
- ISO/IEC 19795-1. (2021). ISO, ISO/IEC 19795-1:2021—Information technology—Biometric performance testing and reporting—Part 1: Principles and framework. Retrieved July, 2021, from <https://www.iso.org/standard/73515.html>.
- ISO/IEC 30107-1. (2016). ISO, ISO/IEC 30107-1:2016—Information technology—Biometric presentation attack detection—Part 1: Framework. Retrieved July, 2021, from <https://www.iso.org/standard/53227.html>.
- ISO/IEC 30107-3. (2017). ISO, ISO/IEC 30107-3:2017—Information technology—Biometric presentation attack detection—Part 3: Testing and reporting. Retrieved July, 2021, from <https://www.iso.org/standard/67381.html>.
- ISO/IEC TR 19795-3. (2007). ISO/IEC, ISO/IEC TR 19795-3:2007—Biometric performance testing and reporting—Technical Report—Modality-specific Testing. ISO/IEC Standard.
- Jain, A. K., Deb, D., & Engelsma, J. J. (2021). Biometrics: Trust, but verify. *IEEE Transactions on Biometrics, Behavior, and Identity Science*. <https://doi.org/10.1109/TBIM.2021.3115465>.
- Jain, A. K., Pankanti, S., Prabhakar, S., Hong, L., Ross, A., & Wayman J. L. (2004). Biometrics: A grand challenge. In *Proceedings of International Conference on Pattern Recognition* (17th ed., Vol. 2, pp. 935–942).
- Jain, A. K., Prabhakar, S., & Pankanti, S. (2002). On the similarity of identical twin fingerprints. *Pattern Recognition*, 35(11), 2653–2663.
- Jain, A. K., Ross, A., & Pankanti, S. (2006). Biometrics: A tool for information security. *IEEE Transactions on Information Forensics and Security*, 1(2), 125–143.
- Lee, H. C., & Gaensslen, R. E. (2012). *Advances in fingerprint technology* (3rd ed.). CRC Press.
- Levi, G., & Sirovich, F. (1972). Structural description of fingerprint images. *Information Sciences*, 327–355.
- Lin, C. H., Liu, J. H., Ostenberg, J. W., & Nicol, J. D. (1982). Fingerprint comparison I: Similarity of fingerprints. *Journal of Forensic Sciences*, 27(2), 290–304.
- Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L., & Jain, A. K. (2000, September). *FVC2000: Fingerprint verification competition*. Technical Report: DEIS, University of Bologna.
- Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L., & Jain, A. K. (2002). FVC2002: Second finger- print verification competition. In *Proceedings of International Conference on Pattern Recognition* (16th ed.).
- Moenssens, A. (1971). *Fingerprint techniques*. Chilton Book Company.
- Monson, K. L., Roberts, M. A., Knorr, K. B., Ali, S., Meagher, S. B., Biggs, K., Blume, P., Brando- delli, D., Marzoli, A., Reneau, R., & Tarasi, F. (2019). The permanence of friction ridge skin and persistence of friction ridge skin and impressions: A comprehensive review and new results. *Forensic Science International*, 297, 111–131.
- Mordor. (2021). Mordor intelligence. Consumer biometrics market—growth, trends, COVID-19 impact, and forecasts (2022–2027). Retrieved July, 2021, from <https://www.mordorintelligence.com/industry-reports/consumer-biometrics-market>.
- Mulvihill, J. J. (1969). The genesis of dermatoglyphics. *The Journal of Pediatrics*, 75(4), 579–589.
- Penrose, L. S. (1965). Dermatoglyphic topology. *Nature*, 205, 545–546.
- Phillips, P. J., Martin, A., Wilson, C. L., & Przybocki, M. (2000, February). An introduction to evaluating biometric systems. *IEEE Computer Magazine*, 33(2), 56–63.
- Ratha, N. K., Connell, J. H., & Bolle, R. M. (1999). A biometrics-based secure authentication system. In *Proceedings of Workshop on Automatic Identification Advances Technologies*.
- Ratha, N. K., Connell, J. H., & Bolle, R. M. (2001). An analysis of minutiae matching strength. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (3rd ed., pp. 223–228).

- Ratha, N. K., Connell, J. H., & Bolle, R. M. (2003). Biometrics break-ins and band-aids. *Pattern Recognition Letters*, 24(13), 2105–2113.
- Rhodes, H. T. F. (1956). *Alphonse Bertillon: Father of scientific detection*. Abelard-Schuman.
- Roddy, A., & Stosz, J. (1997). Fingerprint features: Statistical-analysis and system performance estimates. *Proceedings of the IEEE*, 85(9), 1390–1421.
- Ross, A. A., Nandakumar, K., & Jain, A. K. (2006). *Handbook of multibiometrics*. Springer Science & Business Media.
- Scott, W. (1951). *Fingerprint mechanics—A handbook*. C. Thomas, Springfield.
- Singh, M., Singh, R., & Ross, A. (2019). A comprehensive overview of biometric fusion. *Information Fusion*, 52, 187–205.
- Viveros, R., Balasubramanian, K., & Mitas, J. A. (1984). Binomial and negative binomial analogues under correlated bernoulli trials. *Journal of the American Statistician*, 48(3), 243–247.
- Watson, C. I., & Wilson, C. L. (1992). *NIST special database 4, fingerprint database*. U.S. National Institute of Standards and Technology.
- Wayman, J. L. (1999). Fundamentals of biometric authentication technologies. In J. L. Wayman (Ed.), *National biometric test center collected works* (Vol. 1). National Biometric Test Center.
- Wayman, J. L. (2001). Confidence interval and test size estimation for biometric data. *Personal Communication*.



Fingerprint Sensing

2

Abstract

This chapter surveys available fingerprint acquisition techniques: from the traditional “ink on paper” to live-scan sensing based on optical, capacitive, and ultrasonic technologies. Technological advancements (e.g., the TFT process) that enabled in-display integration of fingerprint sensors in smartphones and emerging technologies such as OCT and touchless sensing are introduced. Examples are provided for multi-finger and single-finger scanners, as well as sensing elements for mobile devices. Factors that determine the quality of the sensing device and the resulting fingerprint image are explained, and the most common Image Quality Specifications (IQS) used for sensor certification are reviewed.

Keywords

Fingerprint reader • Live-scan sensing • Sensing technologies • Image quality specification • Compression

2.1 Introduction

Historically, in law enforcement applications, the acquisition of fingerprint images was performed by using the so-called “ink technique”: the subject’s fingers were smeared with black ink and pressed or rolled on a paper card; the card was then scanned by using a general-purpose scanner, producing a digital image. This kind of acquisition process is referred to as *off-line* fingerprint acquisition or off-line sensing and is briefly discussed in Sect. 2.2. A particular case of off-line sensing is the acquisition of latent fingerprints from crime scenes (Colins, 1992). Nowadays, almost all civil and criminal AFIS accept *live-scan* digital images acquired by directly sensing the finger surface with an electronic

fingerprint scanner (also called *fingerprint reader*). No ink is required in this method, and all that a subject has to do is to present his finger to a live-scan scanner. Although AFIS has greatly benefited from the use of live-scan acquisition techniques, this innovation is undoubtedly more important for a broad range of civil, commercial, and personal applications where user acceptance and convenience, low cost, and reliability are necessary and expected. Figure 2.1 provides an overview of the fingerprint scanner evolution over the last 30 years.

The general structure of a typical fingerprint scanner is shown in Fig. 2.2: a *sensor* reads the ridge pattern on the finger surface and converts the analog reading to a digital form through an A/D (Analog to Digital) converter, where an interface module is responsible for communicating (sending images, receiving commands, etc.) with external devices (e.g., a personal computer). Throughout this chapter, we use the terms “scanner” and “sensor” with different meanings: with *sensor*, we denote the internal active sensing element of a fingerprint scanner that reads the finger surface.

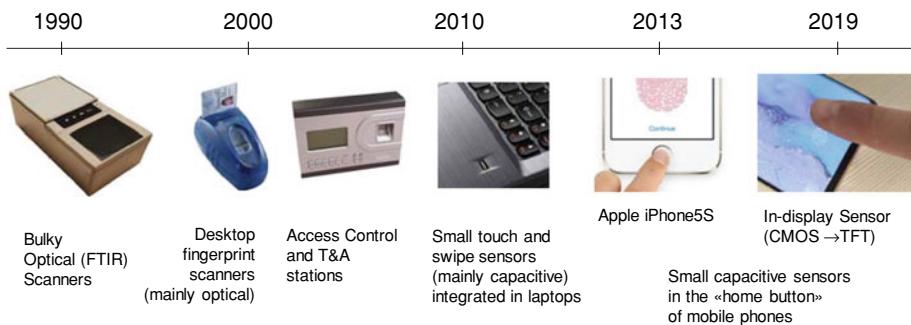


Fig. 2.1 Fingerprint scanners evolution over the last 30 years. In Sect. 2.3, details are provided for the technologies shown here

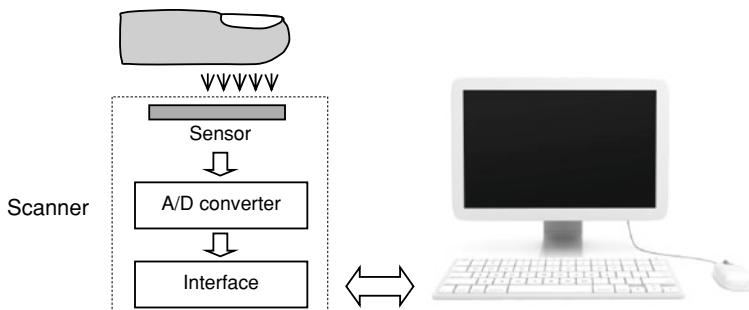


Fig. 2.2 Block diagram of a fingerprint scanner connected to a personal computer

The different technologies the sensors are based on (e.g., optical, capacitive, and ultrasound) are surveyed in Sect. 2.3. In practice, there exist several variants of the schema in Fig. 2.2: for example, often the sensor output is already a digital signal, and therefore no separate A/D conversion is necessary; some fingerprint scanners may not have an integrated A/D converter and an external frame grabber would be needed to transform their analog output signal. Furthermore, some embedded System-on-a-Chip devices have been proposed (e.g., Jung et al., 2005) where, besides the sensor, a processing board is embedded into the chip to locally process and/or match the fingerprint data (see Sect. 9.7). The design of secure fingerprint-based biometric systems requires protection/encryption mechanisms to be implemented in the biometric scanners. Chapter 9 discusses the techniques used to protect fingerprint scanners against various attacks and to detect fake fingers presented to the sensors.

Existing scanners can be classified into one of the following categories:

- *Multi-finger*: more than one finger can be acquired simultaneously (see Fig. 2.3a and b). Usually, the four fingers of one hand (all except the thumb) can be acquired at the same time so that three acquisitions are enough to capture all the 10 fingers in the sequence: four fingers (first hand), four fingers (second hand), and the two thumbs together. A typical usage of multi-finger scanners is in forensic and other large-scale civil applications where more than one finger is used to enroll and identify individuals. The segmentation of a single image containing four fingerprints into four separate single fingerprints is known as *slap segmentation* (see Fig. 2.25); this task is generally performed in software (Gupta & Gupta, 2012, 2014, 2016; Zhang et al., 2010). NIST organized evaluation campaigns to determine the accuracy of slap segmentation algorithms; the results of the recent Evaluation III are reported in (NIST-Slap 2019¹).
- *Single-finger*: only one finger at a time can be acquired (see Fig. 2.3c); this type of scanner is most widely used in commercial and personal applications due to its small size, low cost, and simplicity of use. Designing compact and cheap scanners is crucial to allow fingerprint scanners to be embedded in low-cost, portable devices such as laptops and smartphones. Swipe sensors (discussed in Sect. 2.4) are tiny sensing elements that acquire the image of just a small portion of the finger at a time and require the user to sweep the finger over the sensor surface; however, their adoption is being abandoned because of usability problems. To achieve reliable fingerprint recognition, fingerprint images should possess certain characteristics. Overly relaxing some of the constraints (e.g., image quality and minimum finger area) may lead to a significant (and sometimes unacceptable) decrease in fingerprint recognition accuracy. Fortunately, as discussed in Sect. 2.3, a number of technologies are being developed to manufacture large-area sensors that can be embedded in the display of a mobile phone or a tablet.

Examples of both multi-finger and single-finger scanners are included in Sect. 2.8.

¹ <https://www.nist.gov/itl/iad/image-group/slap-fingerprint-segmentation-evaluation-iii>.

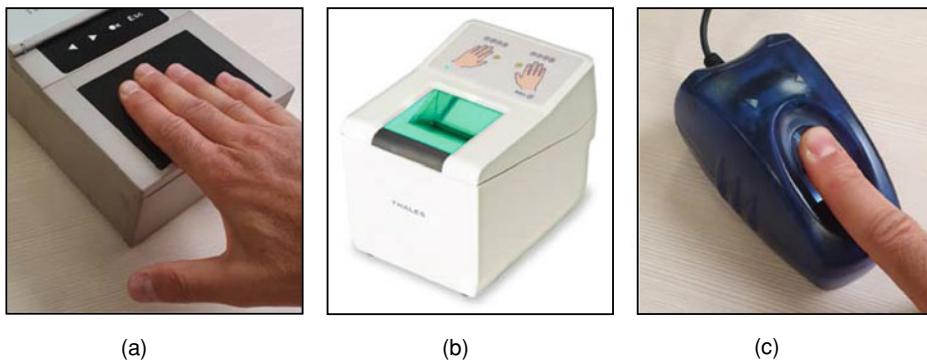


Fig. 2.3 Fingerprint scanners. **a** simultaneous acquisition of four fingers through a multi-finger scanner; **b** dual-finger/rolled scanner; **c** acquisition with a single-finger scanner

Some scanners (see Fig. 2.3b) can also acquire *rolled impressions*: the user is required to roll a finger “nail-to-nail” on the scanner, thus producing an unwrapped representation of the whole fingerprint pattern which carries more information with respect to a *flat* (also called *dab*, *slapped*, or *plain*) impression (see Fig. 2.4). Rolled impressions are particularly useful when populating a criminal fingerprint database because the searched latent fingerprints taken from crime scenes are often partial or off-center, showing the sides of a fingertip or fragments that only a rolled impression can capture.

It is often necessary for a trained fingerprint acquisition expert to assist the user in rolling his or her finger on the sensor. Hardware and software techniques have been introduced to enable live-scan fingerprint scanners, operating at proper frame rate, to



Fig. 2.4 The same finger acquired as a plain impression (on the left) and as a rolled impression (on the right); the portion of the rolled fingerprint corresponding to the plain fingerprint is highlighted

compose the sequence of images of the rolling finger into a single unwrapped impression (see, for example, Ratha et al., 1998; Kwon et al., 2010). A Nail-to-Nail (N2N) Fingerprint Challenge was organized by IARPA in 2017 (see Fiumara et al., 2017) to improve rolled fingerprint capture technology without requiring the presence of a human operator.

To maximize the compatibility between fingerprint images acquired by different scanners and ensure good quality of the acquired image, specifications have been released by the FBI and other organizations that are involved in large-scale biometric deployments. In fact, as demonstrated by Ross and Jain (2004), Han et al. (2006), Ross and Nadgir (2006, 2008), Jang et al. (2007), Modi et al. (2009), Jia et al. (2012), Lugini et al. (2013), Marasco et al. (2013), Lazarick and Wolfhope (2016), and AlShehri et al. (2018), matching images of the same finger acquired by different scanners, not compliant with given specifications, can lead to a severe drop in fingerprint recognition accuracy. This problem can be partially alleviated with a priori calibration techniques, as shown in Ross and Nadgir (2008), where an average spatial compensation model is a priori determined for two given fingerprint scanners. However, the characteristics of different devices of exactly the same model/manufacturer can markedly deviate with respect to their nominal values; therefore, a specific compensation would be required for any pair of two devices, and this is unfeasible in medium–large scale applications. Even minor changes in the sensor resolution (resulting in a different image scale) can lead to an accuracy drop: Zang et al. (2013) extended the MCC minutiae-matcher (see Sect. 4.4.3) by explicitly considering the scale and reported improved scanner interoperability.

Finally, while geometric calibration can improve interoperability for minutiae-based matching, textural features and small ridge details can look quite different in fingerprint images captured by different scanners and ad hoc enhancement techniques could be necessary for feature-based matching (AlShehri et al., 2018).

Section 2.5 introduces the parameters of fingerprint images and Sect. 2.6 summarizes the FBI specifications. Section 2.7 defines the scanner quality from an operational point of view that makes explicit the relationship between the quality parameters of fingerprint scanners and the performance of fingerprint recognition algorithms.

Matching fingerprints acquired through small-area sensors is a difficult task due to the possibility of having too little overlap between different acquisitions of the same finger: Sect. 2.9 discusses this problem and proposes some techniques that can partially solve it.

Most of the fingerprint recognition systems, especially those in commercial applications, do not store fingerprint images but store only numerical features extracted from the image (see Chap. 3). However, in certain applications (e.g., law enforcement and electronic documents), it may be necessary to store the fingerprint images acquired during enrollment in a database so that a trained expert can verify the matching results output by an AFIS. Storing millions of fingerprint images (as in a large AFIS), or transmitting these images through low-bandwidth networks, is particularly demanding in terms of space/time. Hence, ad hoc compression techniques have been proposed; Sect. 2.10 briefly discusses fingerprint image compression.

2.2 Off-Line Fingerprint Acquisition

Although the first fingerprint scanners were introduced more than 40 years ago, the ink technique (Lee & Gaenslen, 2012; Reed & Meier, 1990) is still used in law enforcement applications. The use of ink techniques has gradually been replaced by live-scan acquisition techniques. As a result, the databases that have been built by law enforcement agencies over a long period of time (tens of years) contain fingerprint images acquired by both off-line as well as live-scan scanners. The AFIS fingerprint recognition algorithms are expected to interoperate on these different types of images. In other words, an image acquired using an off-line scanner needs to be matched to an image acquired using live-scan scanners without any loss of accuracy. As mentioned earlier, in the ink technique, the finger skin is first smeared with black ink, pressed or rolled against a paper card, and converted into digital form by means of a paper scanner. Images acquired with the ink technique are also denoted as *inked fingerprints*. The most commonly used resolution of the scanner/camera is 500 dpi. The ink technique often produces images that include regions with missing fingerprint information due to excessive or insufficient ink on the finger or excessive or insufficient finger pressure. Figure 2.5 shows two examples of digitized images from fingerprint cards. These images have been taken from the NIST Special Database 14 (Watson, 1993).

In forensic applications, a special kind of fingerprint image, called *latent fingerprint* (or *fingermark*), is of great interest. These are partial fingerprint images lifted from a crime scene that are used to apprehend suspects and convict criminals. Constant perspiration exudation of sweat pores on fingerprint ridges and intermittent contact of fingers with other parts of the human body and various objects leave a film of moisture and/or grease on the surface of the fingers. When touching an object (e.g., a glass surface), the film of moisture and/or grease is transferred from the finger to the object and leaves an



Fig. 2.5 Two rolled fingerprint images acquired off-line with the ink technique

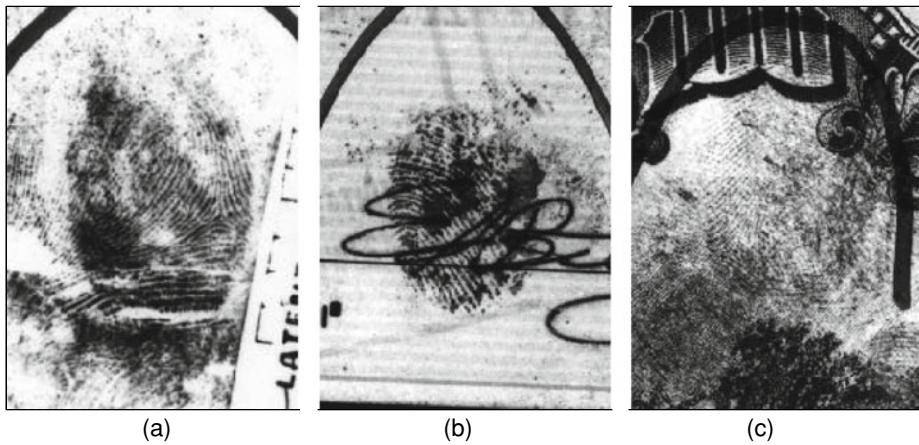


Fig. 2.6 Examples of **a** good, **b** bad, and **c** ugly latent fingerprints from NIST Special Database 27 (Garris & McCabe, 2000)

impression of the ridges on the object thereon. In this case, the actual finger that left the impression on the object is not available, so a copy of the latent print needs to be lifted from the surface of the object. Latent fingerprints are not clearly visible, and their detection often requires some means of chemical development and enhancement (Fig. 2.6). Powder dusting, ninhydrin spraying, iodine fuming, and silver nitrate soaking are four commonly used techniques of latent print development (Lee & Gaenssen, 2012). These techniques are quite effective under normal circumstances but are not appropriate in special cases when fingerprints are deposited on certain objects or surfaces (e.g., wet surfaces, untreated wood, and human skin). Better procedures have been developed based on new chemical reagents, instruments, and systematic approaches involving a combination of methods (Champod et al., 2016; Lee & Gaenssen, 2012) to develop latent fingerprints from such surfaces.

2.3 Live-Scan Fingerprint Sensing

The most important part of a live-scan fingerprint scanner is the sensor (or sensing element), which is the component where the fingerprint image is formed. Most of the fingerprint scanners manufactured before 10 years ago were based on optical sensors equipped with prisms and lenses or solid-state capacitive devices where a silicon platen was put in direct contact with the fingertip. While the main drawback of optical sensors was the large size, manufacturing large-area solid-state sensors was critical and expensive. Solid-state sweep sensors were introduced as an alternative approach to reduce silicon

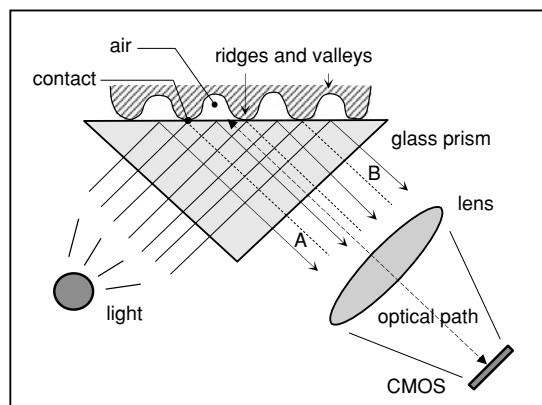
area; however, due to their scarce usability, such devices never reached commercial success. In the last decade, the strong demand from the mobile phone sector, where biometric authentication is now pervasive, led to the development of alternative designs. Instead of using a silicon wafer, the Thin-Film Transistor (TFT) process (Lu et al., 2018) is used to deposit an array of tiny detectors on a (transparent) glass or plastic substrate: optical, capacitive, and ultrasound sensing technologies have been redesigned to work in this setting that also allows embedding the sensor in the display of a smartphone or on the surface of a flexible payment card. The sensing area limitation of solid-state sensors is no more a concern, and full-display fingerprint sensing capabilities have been already announced for the next generation of smartphones.

2.3.1 Optical Sensors

- *Frustrated Total Internal Reflection (FTIR):* FTIR (Hase & Shimisu, 1984) is the oldest and still the most commonly used technique in forensic and government applications. As the finger touches the top side of a glass/plastic prism, the ridges are in optical contact with the prism surface, but the valleys remain at a certain distance (see Fig. 2.7). The left side of the prism is typically illuminated through a diffused light (a bank of light-emitting diodes [LEDs] or a film of planar light). The light entering the prism is reflected at the valleys, and randomly scattered (absorbed) at the ridges. The lack of reflection allows the ridges (which appear dark in the image) to be discriminated from the valleys (appearing bright). The light rays exit from the right side of the prism and are focused through a lens onto a CMOS image sensor. Because FTIR devices sense a three-dimensional finger surface, they cannot be easily deceived by the presentation of a photograph or printed image of a fingerprint.

The FTIR-based sensor shown in Fig. 2.7 often introduces certain geometrical distortions. The most evident one is known as trapezoidal or keystone distortion and is

Fig. 2.7 FTIR-based fingerprint sensor operation



produced by the perspective view of the imaging surface. Since the fingerprint plane is not parallel to the CMOS plane, rays A and B in Fig. 2.7 have different lengths, resulting in a stretching or compression of the image regions which is a function of their distance from the optical axis. Compensation for this distortion may be optics-based or software-based (i.e., calibration techniques).

When a finger is very dry, it does not make uniform and consistent contact with the FTIR imaging surface. To improve the formation of fingerprints from dry fingers, whose ridges do not contain sufficient sweat particles, some scanner producers use conformal coating (typically made of silicone), which improves the optical contact of the skin with the prism. With the aim of reducing the cost of optical devices, plastic is often used instead of glass for prisms and lenses.

In spite of generally superior image quality and potentially larger sensing areas, FTIR-based fingerprint devices cannot be miniaturized unlike other techniques, especially in thickness. In fact, the length of the optical path (i.e., the distance between the prism external surface and the image sensor) cannot be significantly reduced without introducing severe optical distortion at the edges of the image; using intermediate mirrors or a sheet prism (Zhou et al., 1998) typically helps in assembling working solutions in reasonably small packages, but even if these are suitable for embedding into a mouse or a keyboard, they are still too large to be integrated into a mobile phone.

- *Lensless design:* A significant reduction of the packaging size can be achieved by removing prism and lens and tightly coupling an array of photodetectors to the inner side of the platen. To this purpose, two main problems need to be solved: (i) the light photons reflected from the finger need to be guided on the photodetectors without crosstalk between adjacent pixels; (ii) since the magnification effect of a lens cannot be exploited, the photodetector array must be large as the whole sensing area, and this would result in a high cost if CMOS imaging was used.

The first problem can be solved by using a fiber optic layer or a light collimator (see Fig. 2.8). Another approach is illuminating the finger by using cone-shape light with acute incidence angle, as proposed in (Bae et al., 2018); see Fig. 2.9.

The second problem can be addressed by replacing the CMOS with a TFT sensor. In the TFT process, which is nowadays a mature and inexpensive technology for the

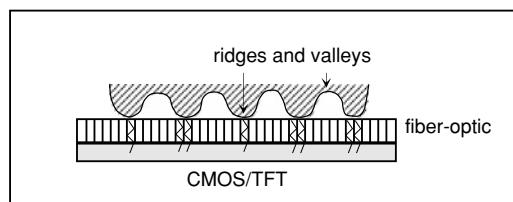


Fig. 2.8 A sensor based on optical fibers. Residual light emitted by the finger is conveyed through micro-optical guides to the array of photodetectors that constitute the CMOS or TFT backplane

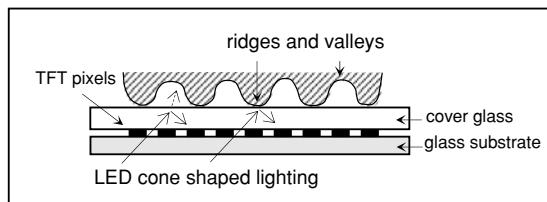
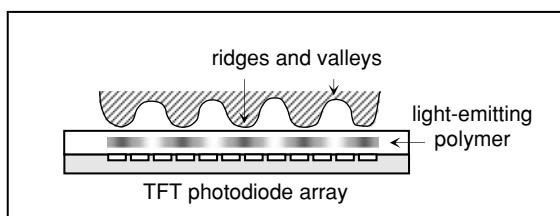


Fig. 2.9 A lensless optical sensor based on TFT technology. In the Hidden Optical Touch configuration proposed by Bae et al. (2018), the LED light source is placed under the TFT glass substrate and is partially masked by the pixel array. A cone shape light with a proper incidence angle reaches the cover glass surface and is reflected back to the photodetectors without crosstalk between pixels

fabrication of large LCD panels, transistors are created by depositing a thin film of amorphous silicon on a glass substrate. Transistors cover only a small fraction of the area of each pixel, and the rest of the film is etched away to allow light to easily pass through. To develop a TFT optical sensor, each pixel of the array is composed of a photodiode and readout transistor. The photodiode can be built with amorphous silicon (Bae et al., 2018) or printed organic materials (Tordera et al., 2019). TFT technology allows designing large-area panels (Liao et al., 2015) and high-resolution devices (Huang et al., 2015). Transparent materials and flexible substrates can be employed to embed TFT optical sensors in the display of portable devices or to wrap them around curved surfaces.

- *Electro-luminescent*: an electro-luminescent panel contains a polymer that, when polarized with the proper voltage, emits light that depends on the potential applied on one side. Since ridges touch the polymer and the valleys do not, the potential is not the same across the surface when a finger is placed on it; the amount of light emitted varies, thus allowing a luminous representation of the fingerprint pattern to be generated (see LES technology by Integrated Biometrics, 2019). The luminescent pattern can be converted into a digital image with a conventional CMOS camera or a TFT photodiode array (see Fig. 2.10).
- *Touchless acquisition*: a high-quality camera with proper lighting (Wang et al., 2009) is used to directly focus on the fingertip. The finger is not in contact with any surface, but the scanner may be equipped with a mechanical support to facilitate the user in presenting the finger at a uniform distance. Touchless acquisition (Parziale, 2007)

Fig. 2.10 Electro-optical fingerprint sensor



may be perceived to be more hygienic and may overcome some problems of touch-based acquisition such as the non-linear distortion caused by pressing the finger against the sensor platen and the need for periodically cleaning the sensor surface; however, obtaining well-focused and high-contrast images is still quite challenging with the touchless methods, and a recent NIST study confirms that accuracy is inferior with respect to devices requiring physical contact (Libert et al., 2019). Nevertheless, fully touchless acquisition (i.e., with no mechanical guides) is currently receiving a lot of interest for applications where multiple users have to interact with the same device (e.g., border control), because of the lower risk of Covid-19 transmission.

Fingerprint images acquired with a smartphone camera are often denoted as *finger-photos* (see Fig. 2.11) or *finger-selfies* (Malhotra et al., 2019). Hiew et al. (2007) used an 8-megapixel digital camera as an input device for their system, while Lee et al. (2006, 2008), Stein et al. (2012), and Sankaran et al. (2015) dealt with the problem of selecting and processing fingerprint images acquired with the camera integrated into a mobile device.

A three-dimensional fingerprint representation (Kumar, 2018) can be obtained with touchless devices by using multiple cameras/views (Parziale et al., 2006; Choi et al., 2010; Liu et al., 2013; Liu & Zhang, 2014; Donida Labati et al., 2016), structured light (Wang et al., 2010), and a single camera with different illuminations (Kumar & Kwong, 2015) or laser scanning (Galbally et al., 2017). An example of a three-dimensional fingerprint is shown in Fig. 2.12. A rolled-equivalent impression can be produced by a 3D→2D unwrapping (Chen et al., 2006; Fatehpuria et al., 2006), or the native three-dimensional features can be directly used for fingerprint matching (Kumar & Kwong, 2015; Yin et al., 2021).

An interesting touchless device (called MorphoWaveTM; see Fig. 2.13) has been proposed by the company IDEMIA for access control and border control applications where high throughput and hygiene are relevant issues.

- *Multispectral imaging*: multispectral sensors capture multiple images of the same finger using different wavelengths of light, different illumination orientations, and different

Fig. 2.11 A fingerprint (acquired with an optical FTIR scanner) and a fingerphoto (acquired with smartphone camera) of the same finger: the ridge-valley contrast is typically much higher in contact acquisitions



Fig. 2.12 A three-dimensional fingerprint representation obtained with a structured light approach

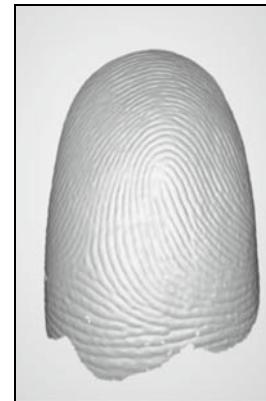


Fig. 2.13 MorphoWave™ scanner by IDEMIA (formerly Safran Morpho). The four fingerprints of one hand are acquired on the fly while the user swipes his hand through the device aperture without touching any surface. The device uses multiple cameras to acquire an intermediate three-dimensional representation



polarization conditions (Rowe et al., 2007). The resulting data can be processed to generate a single composite fingerprint image. Multispectral imaging is considered more robust than other acquisition techniques when fingerprint images are captured under adverse influences such as suboptimal skin condition and bright ambient light (Rowe & Nixon, 2005; Rowe et al., 2007). Furthermore, features extracted from multispectral images are better suited in discriminating between real and fake fingers. On the other hand, multispectral imaging devices are more complex and expensive than conventional optical scanners and their adoption is still limited.

Engelsma et al. (2019) proposed Raspireader, an open-source fingerprint scanner capable of simultaneously acquiring an FTIR image and a direct image. Using both of these image streams, complementary information can be exploited to improve the fingerprint quality and make presentation attack detection simpler.

- *Optical Coherence Tomography (OCT):* optical coherence tomography uses low-coherence light to capture depth images of biological tissues. While its main application is in the medical field, it has been proved (Auksorius & Boccaro, 2015, 2017;

Darlow & Connan, 2015; Aum et al., 2016; Sousedik & Breithaupt, 2017; Auksorius et al., 2020) that fingerprint images can be effectively acquired with OCT devices. The main advantage of OCT imaging is that the inner “copy” of the epidermis pattern (called dermis) can be detected (see Fig. 2.14), and when the skin surface is damaged, the dermis pattern can provide more reliable information (see Fig. 2.15). Furthermore, sub-surface information, such as papillary junctions and sweat glands, can be exploited for presentation attack detection (Chugh & Jain, 2019). However, current OCT devices are still bulky, slow, expensive, and not mature for commercial applications.

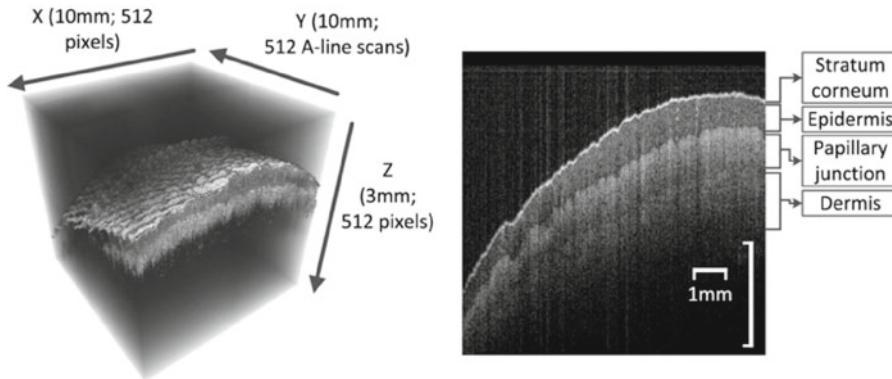


Fig. 2.14 (Left) Example of an OCT scan volume and (Right) one of its cross-sections where different skin layers are highlighted. © The Optical Society. Reprinted, with permission, from Darlow and Connan (2015)

Fig. 2.15 (Left) the FTIR image of a finger whose epidermis was voluntarily erased by one hour of sandpaper rubbing and (Right) the finger in the same condition acquired with an OCT device. Image courtesy of Christophe Champod—University of Lausanne



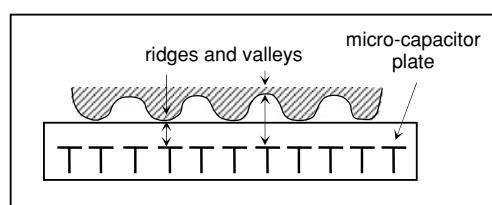
2.3.2 Capacitive Sensors

Although capacitive sensors have been proposed in patent literature since the 1980s (Edwards, 1984; Tsikos, 1982), it was not until the mid-1990s that they became commercially viable (Xia & O’Gorman, 2003). A capacitive sensor is a two-dimensional array of micro-capacitor plates embedded in a panel (see Fig. 2.16). The other plate of each micro-capacitor is the finger skin itself. Small electrical charges are created between the surface of the finger and each of the silicon plates when a finger is placed on the chip. The magnitude of these electrical charges depends on the distance between the fingerprint surface and the capacitance plates (Tartagni & Guerrieri, 1998). Thus, fingerprint ridges and valleys result in different capacitance patterns across the plates. An accurate capacitance measurement is quite difficult to make and adjust, and each vendor has its own method to get enough sensitivity to make a difference between the ridges and the valleys. The capacitive sensors, like the optical ones, cannot be easily deceived by the presentation of a flat photograph or printed image of a fingerprint since they measure the distances and therefore only a three-dimensional surface can be sensed.

In the first generation of capacitive sensors, the micro-capacitor array was embedded in a silicon chip manufactured with CMOS technology (Inglis et al., 1998; Lee et al., 1999; Morimura et al., 2000); in such a setup, the fingerprint sensing area is limited by the chip size and to keep the cost low only small sensors have been manufactured. In 2013, Apple iPhone5S was one of the first mobile devices to integrate a small capacitive fingerprint sensor in the home button. Shortly thereafter, other smartphone manufacturers integrated capacitive sensors in their devices making biometric authentication the default modality for device unlock. However, as discussed in Sects. 2.7 and 2.9, verification accuracy significantly drops when fingerprint area decreases, and some researchers pointed out potential vulnerabilities related to the use of such small sensors in mobile devices (Roy et al., 2017).

The current generation of capacitive sensors is based on the TFT process (Hashido et al., 2003; Hwang et al., 2017; Jeon et al., 2019; Seo et al., 2018; Young et al., 1997) where the detector array is embedded on a glass substrate. As per the lensless optical sensors, this overcomes the limited sensing area problem and allows in-display integration in mobile devices (Jeon et al., 2016). In general, the design of TFT capacitive sensors is simpler than optical TFT ones because there is no need to provide illumination from

Fig. 2.16 Capacitive sensing



inside and to guide the reflected photon beams on single pixels. However, a disadvantage of capacitive technology is the higher difficulty to acquire image information through thick layers (e.g., the protective glass of a mobile device display).

A critical component (especially in the first generation of capacitive sensors) is the surface coating: the silicon chip needs to be protected from chemical substances (e.g., sodium) that are present in finger perspiration. But a coating that is too thick increases the distance between the pixels and the finger, lowering the ability to discriminate between a ridge and a valley, especially for poor-quality fingers, where the depth of a valley is in the range of a micron. As a result, the coating must be as thin as possible (a few microns), but not too thin, as it will not be resistant to mechanical abrasions. Also, since the capacitive sensors sense the electrical field, Electrostatic Discharges (ESD) from the fingertip can cause large electrical fields that could severely damage the device itself (Wei et al., 2017). Therefore, proper protection and grounding are necessary to avoid ESD, chemical corrosion, and physical scratches to the sensor surface (Thomas & Bryant, 2000; Setlak et al., 2000). Apple used sapphire glass to protect the iPhone5S home button; other researchers suggested using a less-expensive epoxy molding compound (Tabei et al., 2016).

Active capacitance (also known as *RF imaging*) is a special arrangement (Setlak, 1999, 2004), where a capacitive sensor is equipped with a drive ring that generates a Radio Frequency (RF) signal and a matrix of active antennas that receives a signal modulated by the capacitance between the skin and each detector. The finger must be simultaneously in contact with both the sensor and the drive ring which is usually manufactured as a metal bezel. In general, higher immunity to parasitic effects can be achieved with respect to conventional (i.e., *passive*) capacitive sensors.

2.3.3 Thermal Sensors

Thermal sensors are made of pyro-electric material that generates current based on temperature differentials (Mainguet et al., 1999; Han & Koshimoto, 2008; Miki & Tsuchitani, 2017). The fingerprint ridges, being in contact with the sensor surface, produce a different temperature differential than the valleys, which are at a distance from the sensor surface. The sensors are typically maintained at a high temperature by electrically heating them up, to increase the temperature difference between the sensor surface and the finger ridges. The temperature differential produces an image when contact occurs, but this image soon disappears because the thermal equilibrium is quickly reached and the pixel temperature is stabilized. To overcome this problem, two approaches have been proposed:

- Acquiring the fingerprint image through a sweeping method (as explained in Sect. 2.4), in order to continuously change the points of contact of the ridges.

- Providing user-unperceivable heat pulses through the detector array, a technique also known as Active thermal sensing (see LTPS technology by Next Biometrics (2020)²).

With respect to capacitive technology, thermal sensing is not sensitive to ESD, and it can accept a thick protective coating because the thermal information (heat flow) can easily propagate through the coating. However, a thermal sensor typically draws more energy and the acquisition is slower.

2.3.4 Pressure Sensors

Pressure-sensitive sensors have been designed that produce an electrical signal when a force is applied to them. The first pressure sensors were based on *piezoelectric* materials. The sensor surface is made of a non-conducting dielectric material which, on encountering pressure from the finger, generates an electric charge and then a small amount of electric current (this effect is called the piezoelectric effect) can be detected. The strength of the generated current depends on the pressure applied by the finger on the sensor surface. Since ridges and valleys are present at different distances from the sensor surface, they result in different amounts of current. Unfortunately, these materials are typically not sensitive enough to detect the difference; moreover, the protective coating blurs the resulting image. A number of alternative designs (Mainguet, 2020³) have been then introduced to improve sensitivity and reduce cost including Micro-Electro-Mechanical Systems (MEMS) (Sato et al., 2003, 2005), conductive membrane on TFT substrate, and piezoelectric nanowires. However, fingerprint sensors based on pressure never reached full maturity and market penetration.

2.3.5 Ultrasound Sensors

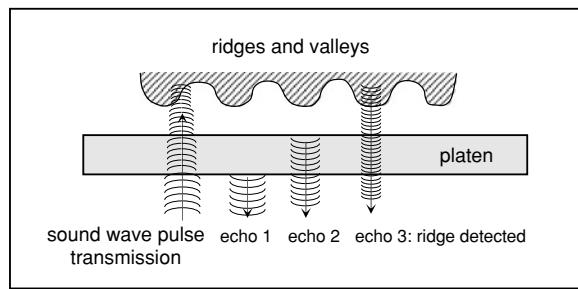
Ultrasound sensing may be viewed as a kind of *echography*. It is based on sending acoustic signals toward the fingertip and capturing the echo signal (see Fig. 2.17). The echo signal is used to compute the range (depth) image of the fingerprint and, subsequently, the ridge structure itself.

An ultrasound sensor has two main components: a transmitter, which generates short acoustic pulses, and a receiver, which detects the responses obtained when these pulses bounce off the fingerprint surface (Bicz et al., 1999; Schneider, 2007; Schneider & Wobischall, 1991). This method images the sub-surface of the finger skin (even through thin

² <https://www.nextbiometrics.com/technology/>.

³ https://mainguet.org/biometrics/types/fingerprint/fingerprint_sensors_physics_mechan.htm#tactile.

Fig. 2.17 The basic principle of the ultrasound technique. A characteristic of sound waves is their ability to penetrate materials, giving a partial echo at each impedance change



gloves); therefore, it is resilient to dirt and oil accumulations on the finger, and can make presentation attack detection easier.

While high-quality images may be obtained by this technology, the first generation of ultrasound scanners were bulky with mechanical parts and quite expensive (several hundred dollars). Moreover, it took a few seconds to acquire an image because of the need for a mechanical scanning.

The current generation of ultrasound sensors is based on two-dimensional arrays of Micromachined Ultrasound Transducer (MUT) that eliminates the need for moving parts (Iula, 2019). The design proposed by Tang et al. (2015, 2016) and Jiang et al. (2017) is based on piezoelectric transducers (PMUT) and the fingerprint sensor can be realized by bonding MEMS and CMOS wafers to achieve compact size, high signal fidelity, and low power dissipation.

The company Qualcomm is currently manufacturing ultrasound sensors for in-display smartphone integration (see 3D Sonic Sensing technology by Qualcomm (2020)⁴). While the current generation of sensors has a small area (e.g., the Samsung Galaxy S10 sensor is about $4 \times 9 \text{ mm}^2$) potentially leading to the security vulnerabilities pointed out by Roy et al. (2017), new ultrasound devices have been already announced with a significantly larger area (e.g., $20 \times 30 \text{ mm}^2$).

2.4 Swipe Sensors

Before the introduction of the TFT process for the creation of detector arrays on glass or plastic substrates, solid-state sensors were manufactured with CMOS technology on silicon wafers, and the chip cost was proportional to the sensing area. In fact, larger die costs more due to fewer dies per wafer and lower yield; furthermore, large dies are more likely to include defects, resulting in many discarded chips.

With the aim of reducing the silicon sensor footprint and cost (up to one order of magnitude according to Xia & O’Gorman, 2003), the sweep sensing method was proposed which requires the user to *sweep* the finger over the sensor. The size of the swipe sensor

⁴ <https://www.qualcomm.com/products/features/fingerprint-sensors>.

$(w \times h)$ is much smaller than touch or area sensors, where h is the height of the sensor and w denotes the width of the sensor. As the finger is swept across this sensor, partial images of the finger are formed. Since the sweeping consists of a vertical movement only, the sensor width (w) should be as wide as a finger; on the other hand, in principle, the height of the sensor (h) could be as low as one pixel. In practice, however, since the finger swipe speed is unknown and it can vary during the sweeping, it is necessary to have a certain degree of overlap between the different fingerprint readings (slices) to effectively combine them to generate a full fingerprint image. Alternative solutions have been proposed where the sensor height is just one or a few pixels and (i) the finger motion is estimated using additional displacement sensors (Clausen, 2007); (ii) a vertically distorted image is acquired and then normalized trying to remove distortion (Lorch et al., 2004).

At the end of the sweep, a single fingerprint image needs to be reconstructed from the slices. This could be done “on the fly” by combining the slices as they are delivered by the sensor (see Fig. 2.18). Morguet et al. (2004) proposed matching two fingerprints directly at the slice level without any image reconstruction: a pair of slices are matched through normalized correlation, and the optimal matching between two complete slice sequences is found by Viterbi search (Viterbi, 1967). Other approaches to image reconstruction from slices can be found in Lee et al. (1999), Zhang et al. (2005, 2006a, b), Habegger et al. (2012), Mathur et al. (2016), and Mardiansyah et al. (2018).

When compared with touch silicon-based sensors, the most important advantages of the sweep sensors are that they are smaller and typically cheaper. Another advantage is that the sweep sensors are “self-cleaning”: the finger itself cleans the sensor during usage and no latent fingerprint is left on the sensor. However, there are relevant drawbacks, as reported below.

- In general, sweeping is less intuitive and natural than using a touch-based device. So, a novice user may encounter some difficulties in performing the sweeping properly

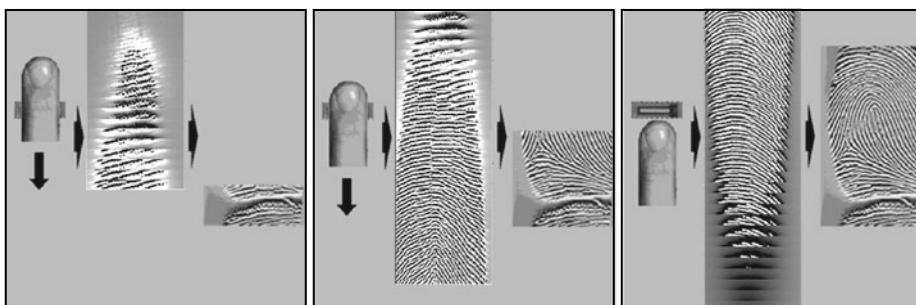


Fig. 2.18 As the user sweeps her finger on the sensor, the sensor delivers new image slices, which are combined to form a two-dimensional image

(i.e., without sharp speed changes or discontinuities). This is supported by a noticeable failure to acquire a rate of 37.9% (Cappelli et al., 2006) for the sweeping sensor during FVC2004 database collection.

- The reconstruction of full fingerprint images from the slices is time-consuming and is prone to errors, especially in the case of poor-quality fingerprints and non-uniform sweep speed.

Therefore, with the advent of TFT sensor manufacturing that allows both miniaturization and low cost, the commercial adoption of swipe sensors fell off.

2.5 Fingerprint Images and Their Parameters

A fingerprint image is a digital representation of a fingerprint pattern acquired through a scanner. The image sampling and quantization processes cause an alteration of the original pattern: the smaller the alteration, the higher the quality of the fingerprint image in terms of fidelity with respect to the original pattern. It is not easy to precisely define the quality of a fingerprint image, and it is even more difficult to decouple the fingerprint image quality from the intrinsic finger quality or condition of the finger. In fact, when the ridge prominence is poor (especially for users engaged in heavy manual work and elderly people), the fingers are too moist or too dry, or fingers are improperly placed, most of the scanners produce poor-quality images (Fig. 2.19). This section focuses on the contribution of the acquisition device to the quality of the image; in Sect. 3.11, the fingerprint image quality is discussed more in general and independently of scanner and user-related factors.



Fig. 2.19 Examples of fingerprint images acquired with an optical scanner: **a** a good-quality fingerprint; **b** a fingerprint left by a dry finger; **c** a fingerprint left by a wet finger; **d** an intrinsically bad fingerprint

The main parameters characterizing the acquisition of a digital fingerprint image, as defined by the FBI (see Appendix F of CJIS (2017)), are as follows:

- *Resolution*: it denotes the number of dots or pixels per inch (*DPI*). A resolution of 500 dpi is the minimum resolution for FBI-compliant scanners and is met by many commercial devices. Figure 2.20 shows the same fingerprint portion sub-sampled at different resolutions; decreasing the resolution results in a greater difficulty in resolving ridges from valleys and isolating minutiae points. A resolution of 250–300 dpi is probably the minimum resolution that allows the fingerprint extraction algorithms to locate the minutiae in fingerprint patterns. Images acquired at 200–300 dpi are often matched through correlation techniques (see Sect. 4.2) which seem to tolerate lower resolutions better (Wilson et al., 2000). 1,000 dpi scanners have started replacing 500-dpi models in forensic applications where analysis of tiny details such as sweat pores, dots, and incipient ridges is very important to match small portions of noisy fingerprint images. Figure 2.21 shows the same fingerprint portion acquired at 1,000 dpi and sub-sampled at 500 dpi. Finally, in some studies, the use of very high-resolution devices (e.g., 1270–3600 dpi) proved to be effective to resolve the fine ridge/valley patterns in newborn, infant, and toddler fingerprints (Jain et al., 2017; Koda et al., 2016; Saggese et al., 2019; Weingaertner et al., 2008). Most of the existing specifications for fingerprint scanners distinguish between *native resolution* and *output resolution*: the former is the actual resolution of the sampling process; the latter is the resolution of the output image that can be adjusted by interpolation or resampling.
- *Area*: this is the size of the rectangular area sensed by a fingerprint scanner. NIST has defined Fingerprint Acquisition Profile (FAP) levels to categorize mobile fingerprint scanners according to their intended use (see Table 3 in NIST (2016) and FAQ in

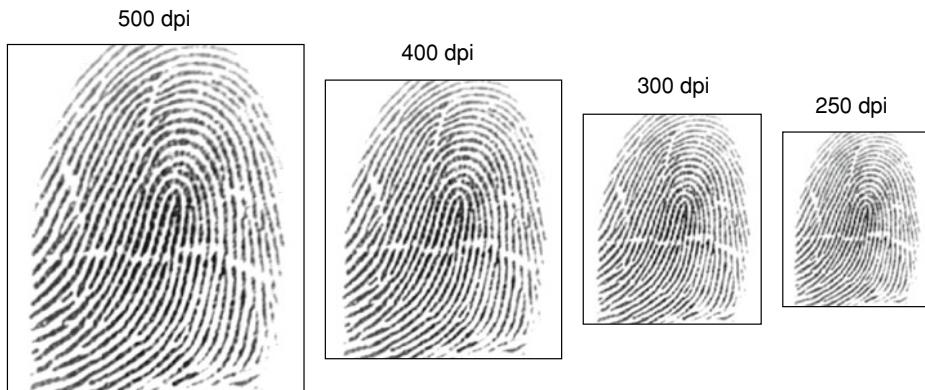


Fig. 2.20 The fingerprint on the left, acquired at 500 dpi, is progressively sub-sampled at lower resolutions: 400, 300, and 250 dpi, respectively

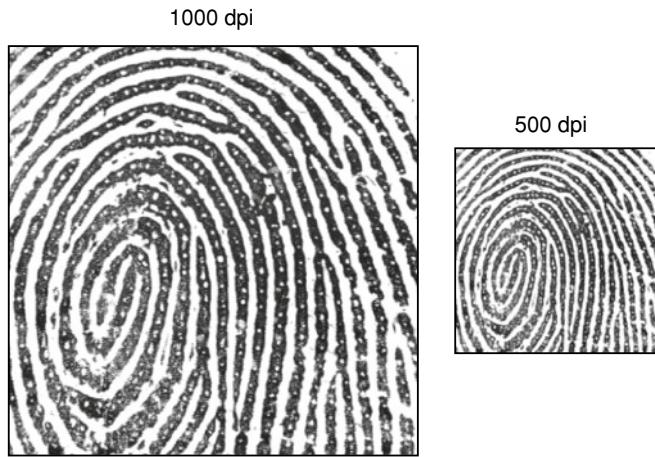


Fig. 2.21 The fingerprint portion on the left is acquired at 1000 dpi; sweat pores and other fine details are clearly visible; on the right, the fingerprint portion is sub-sampled at 500 dpi while the fine details are not as clear

FBI, 2021). The main difference among different FAP levels is the acquisition area as reported in Table 2.1. For multi-finger scanners, the area ($w \times h$) is usually as large as 3.2×2 inch 2 (FAP 50) to allow four fingers to be placed simultaneously; in case of single-finger scanners, an area greater than or equal to 0.8×1 inch 2 (FAP 30) usually permits a full plain fingerprint impression to be acquired. However, in many commercial single-finger scanners, the area is sacrificed to reduce both the cost and the size of the device. Small-area scanners (e.g., \leq FAP 10) do not allow the full fingerprint to be captured, and the users encounter difficulties in consistently presenting the same

Table 2.1 Fingerprint acquisition profile (FAP) levels defined by NIST for mobile fingerprints devices. IQS refer to image quality specifications introduced in Sect. 2.6

FAP level	Sensing area ($w \times h$)	IQS	Fingerprints acquired simultaneously
FAP 10	$0.5'' \times 0.65''$	PIV	1
FAP 20	$0.6'' \times 0.8''$	PIV	1
FAP 30	$0.8'' \times 1.0''$	PIV	1
FAP 40	$1.6'' \times 1.5''$	PIV	2
FAP 45	$1.6'' \times 1.5''$	IAFIS	2
FAP 50	$3.2'' \times 2.0''$	IAFIS	4
FAP 60	$3.2'' \times 3.0''$	IAFIS	4

portion of the finger. This may result in a small overlap between different acquisitions of the same finger, leading to false non-match errors. As discussed in Sect. 2.7, the acquisition area is the most important scanner parameter to maximize the recognition accuracy. Section 2.9 discusses some techniques proposed to deal with small-area sensors.

- *Number of pixels*: the number of pixels in a fingerprint image can be simply derived by the resolution and the fingerprint area: a scanner working at R dpi over an area of $width$ (w) \times $height$ (h) inch 2 has $R \cdot w \times R \cdot h$ pixels. If the area is expressed in mm 2 , the formula must include the mm – inch conversion and, therefore, the number of pixels = $R \cdot (w/25.4) \times R \cdot (h/25.4)$. For example, a scanner working at 500 dpi over an area of 15.24×20.32 mm 2 produces images of $500 \cdot (15.24/25.4) \times 500 \cdot (20.32/25.4) = 300 \times 400$ pixels. The equation is invertible and each value {resolution, area, number of pixels} may be uniquely determined given the other two.
- *Geometric accuracy*: this is usually determined by the maximum geometric distortion introduced by the acquisition device. The geometric distortion can be measured as the absolute value of the difference between the actual distance between two points on a calibrated target and the distance between those same two points as measured on the scanned image of that target. Some of the optical fingerprint scanners introduce geometric distortion which, if not compensated, alters the fingerprint pattern depending on the relative position of the finger on the sensor surface.
- *Gray-level quantization and gray range*: the gray-level quantization denotes the maximum number of gray levels in the output image and is related to the number of bits used to encode the intensity value of each pixel (e.g., 8 bits per pixel yields 256 levels of gray). The gray range is the actual number of gray levels used in an image disregarding the maximum given by the gray-level quantization. Color information is not considered useful for fingerprint recognition, but some researchers have shown that color analysis can be exploited to detect fake fingers (see Sect. 9.5.1).
- *Gray-level uniformity and input/output linearity*: the gray level uniformity is defined as the gray-level homogeneity measured in the image obtained by scanning a uniform dark (or light) gray patch; the Input/Output linearity quantifies the deviation of the gray levels from a linear mapping when the input pattern is transformed into an output image.
- *Spatial frequency response*: the spatial frequency response denotes the ability of an acquisition device to transfer the details of the original pattern to the output image for different frequencies. It is well-known that the fine details corresponding to the high frequencies tend to be smoothed out when a signal is digitally sampled. Spatial frequency response is usually measured through Modulation Transfer Function (MTF) or Contrast Transfer Function (CTF) as explained in Nill et al. (2016); a specific measure for fingerprint scanners, called Top Sharpening Index (TSI), was introduced by Ferrara et al. (2007).

- *Signal-to-noise ratio (SNR)*: the signal-to-noise ratio quantifies the magnitude of the noise with respect to the magnitude of the signal. The signal magnitude is related to the gray range in the output image while the noise can be defined as the standard deviation of the gray levels in uniform gray patches.

Section 2.6 reviews the specifications that the FBI has set for two categories of fingerprint scanners; Sect. 2.7, according to the definition of operational quality, reconsiders the fingerprint image parameters and makes explicit the impact of each of them on the accuracy of automatic fingerprint recognition algorithms.

2.6 Image Quality Specifications for Fingerprint Scanners

In most of the medium- to large-scale biometric applications, the following two requirements are mandatory: (i) the quality of the image must be sufficiently high to guarantee high fingerprint recognition accuracy and (ii) the system must be able to use scanners by different vendors and the fingerprint recognition accuracy should not degrade if the device used for enrollment is different from the one used for recognition. Both these requirements can be fulfilled by enforcing nominal values and tolerances for the parameters defined in Sect. 2.5.

The FBI established an IAFIS⁵ Image Quality Specification (IQS) in order to define the quantitative image quality requirements for IAFIS fingerprint scanners. The IAFIS IQS was defined in Appendix F of the Electronic Fingerprint Transmission Specification (EFTS) (CJIS, 2005), and is also included in the Electronic Biometric Transmission Specification (EBTS) (CJIS, 2017); test procedures to verify compliance of fingerprint scanners to the IQS were delineated in Forkert et al. (1994) and have been revised and updated by Nill et al. (2016). IAFIS IQS-compliant devices typically allow the acquisition of more than one finger at the same time; the minimum sensing area for a plain four-finger scanner is $3.2'' \times 2''$ (see Table 1.1 in Nill et al., 2016). FBI, supported by MITRE experts, certifies devices compliant with the IAFIS IQS: the list of certified devices is maintained in FBI (2021). While IAFIS IQS and related testing procedures were originally conceived for optical-FTIR scanners, the FBI recently evaluated and certified some “non-traditional” scanners (refer to Lazarick & Wolfhope, 2016 and see examples in Table 2.4).

The ISO/IEC 19794-4 (2011) describes the manner in which a fingerprint image must be acquired and stored to maximize interoperability: this document refers to IAFIS IQS as far as the characteristics of fingerprint scanners are concerned. To support the Personal Identity Verification (PIV) program initiated by NIST in 2005 (NIST, 2020), whose goal is to improve the identification and authentication for access to U.S. Federal facilities and information systems, the FBI established a PIV IQS (CJIS, 2006), which defines the quantitative image quality requirements for single-finger capture devices suitable for

⁵ IAFIS is the acronym used to denote the FBI’s Integrated AFIS.

application in the PIV program; these requirements are similar to (but less stringent than) the IAFIS requirements and the corresponding test procedures can be found in Nill (2006).

Table 2.2 summarizes the main IAFIS and PIV IQS requirements. It is worth noting that both IAFIS and PIV specifications have been defined to maximize the fidelity with respect to the original pattern but, as explained in Sect. 2.7, they do not necessarily constitute the best accuracy/cost trade-off when the aim is to maximize the automatic fingerprint recognition accuracy.

2.7 Operational Quality of Fingerprint Scanners

In Sects. 2.5 and 2.6, the scanner quality is defined as “fidelity” in reproducing the original fingerprint pattern. This definition of quality is clearly appropriate to AFIS and other applications where the images may be visually examined by forensic experts. In fact, human experts heavily rely on very fine fingerprint details such as pores and local ridge shape for which high fidelity of the original pattern is fundamental. On the other hand, the requirement is different in automated fingerprint recognition systems, where (i) the images are stored but used only for automated comparisons, or (ii) only fingerprint templates consisting of features derived from the image are stored. As discussed in Cappelli et al. (2008), in these cases it may be more appropriate to define the fingerprint scanner quality as *operational quality*, that is, the ability to acquire images that maximize the accuracy of automated recognition algorithms.

In Cappelli et al. (2008), the impact on the recognition accuracy of the parameters introduced in Sect. 2.5 has been separately assessed through systematic experimentations. In particular:

- The main quality parameters and the corresponding requirements defined by the FBI have been taken into account.
- For each parameter, an approach has been defined to progressively degrade the quality of fingerprint images, thus simulating scanners compliant with gradually relaxed requirements. Figure 2.22 shows some examples of degradation simulation for different parameters.
- Among the four FVC2006 databases (BioLab, 2007), the database acquired with the wider area optical sensor (DB2) has been selected for assessment.
- Ten best-performing algorithms on DB2 from the FVC2006 competition were selected: these algorithms well represent the current state of the art in automated fingerprint recognition technology.
- The correlation between each quality parameter Q and the recognition accuracy was measured by progressively degrading the database images according to Q and analyzing the performance of the 10 algorithms.

Table 2.2 A review of IAFIS and PIV IQS. For the parameter definitions; see Sect. 2.5

Parameter	Requirement	
	IAFIS IQS (4-finger scanners at 500 dpi)	PIV IQS (single-finger scanners)
Area $width (w) \times height (h)$	$w \geq 81.28 \text{ mm (}3.2''\text{)} \text{ and } h \geq 50.8 \text{ mm (}2''\text{)}$	$w \geq 12.8 \text{ mm (}0.504''\text{)} \text{ and } h \geq 16.5 \text{ mm (}0.650''\text{)}$
Native resolution R_N	$R_N \geq 500 \text{ dpi}$	$R_N \geq 500 \text{ dpi}$
Output resolution R_O	$R_O = 500 \text{ dpi} \pm 1\%$	$R_O = 500 \text{ dpi} \pm 2\%$
Gray-level quantization	256 Gray levels (8 bit per pixel)	256 Gray levels (8 bit per pixel)
Gray range DR	for at least 80% of the image: $DR \geq 200$ for at least 99% of the image: $DR \geq 128$	for at least 80% of the image: $DR \geq 150$
Geometric accuracy D_{AC} (ACross-bar) D_{AL} (ALong-bar)	for at least 99% of the test: $D_{AC} \leq 1\%$ $D_{AL} \leq 0.016''$	for at least 99% of the test: $D_{AC} \leq 1.8\%$ $D_{AL} \leq 0.027''$
Gray-level uniformity ^a	for at least 99% of the cases: $D_{RC}^{dark} \leq 1$; $D_{RC}^{light} \leq 2$ for least for 99.9% of the pixels: $D_{PP}^{dark} \leq 8$; $D_{PP}^{light} \leq 22$ for every two small areas: $D_{SA}^{dark} \leq 3$; $D_{SA}^{light} \leq 12$	for at least 99% of the cases: $D_{RC}^{dark} \leq 1.5$; $D_{RC}^{light} \leq 3$ for at least 99% of the pixels: $D_{PP}^{dark} \leq 8$; $D_{PP}^{light} \leq 22$ for every two small areas: $D_{SA}^{dark} \leq 3$; $D_{SA}^{light} \leq 12$
I/O linearity ^b D_{Lin}	$D_{Lin} \leq 7.65$	No requirements
Spatial frequency response	$\text{MTF}_{\min}(f) \leq \text{MTF}(f) \leq 1.05$ see Nill et al. (2016) for IAFIS $\text{MTF}_{\min}(f)$	$\text{MTF}_{\min}(f) \leq \text{MTF}(f) \leq 1.12$ see Nill (2006) for PIV $\text{MTF}_{\min}(f)$
Signal-to-noise ratio ^c SNR	$SNR \geq 125$	$SNR \geq 70.6$

^aDefined as the gray-level differences in a uniform dark (or light) gray patch. Gray-level uniformity is evaluated by dividing the acquisition area in $0.25'' \times 0.25''$ regions and measuring the differences between (i) the average gray levels of adjacent rows/columns D_{RC} ; (ii) the average gray level of any region and the gray level of each of its pixels D_{PP} ; (iii) the average gray levels of any two regions D_{SA}

^b D_{Lin} is measured as the maximum deviation of the output gray levels from a linear least squares regression line fitted between input signal and output gray levels scanning an appropriate target.

^cActually in PIV IQS, this requirement is given by setting the maximum noise standard deviation to 3.5. To make it comparable with the corresponding IAFIS IQS, here this value is transformed to SNR under the hypothesis of a 247 gray-level range (Nill et al., 2016): $\text{SNR} = 247/3.5 = 70.6$.

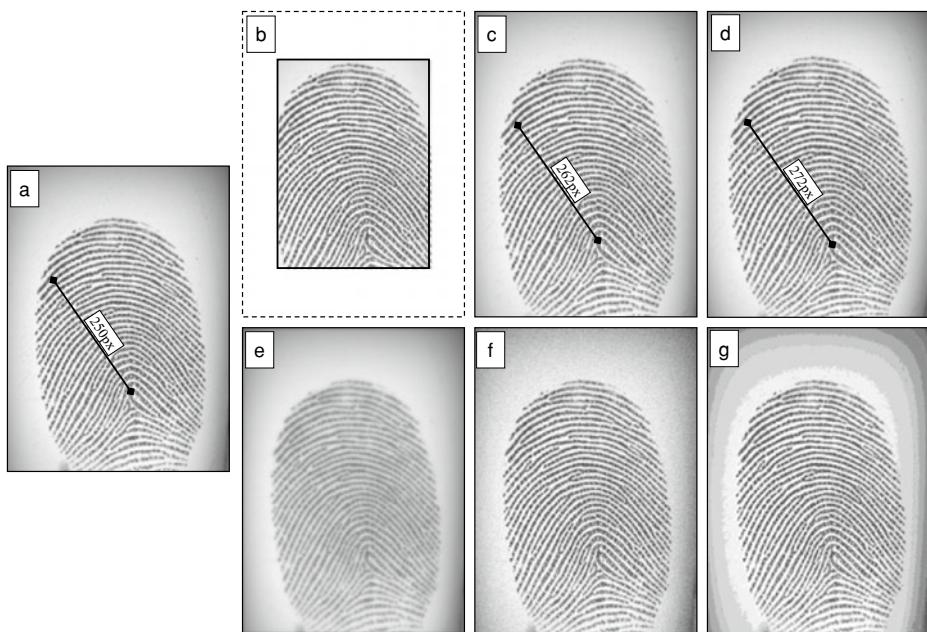


Fig. 2.22 Some examples of image degradations: **a** original image; **b** reduction of area; **c** change of resolution; **d** barrel distortion; **e** reduction of spatial frequency response; **f** reduction of SNR; **g** reduction of gray range

- Results are reported in terms of the percentage performance drop: let EER_O be the original equal error rate (without any degradation) and EER_D be the equal error rate measured after applying a degradation D , then the performance drop is computed as $(EER_D - EER_O)/EER_O$.

The results of the experiments lead to the following conclusions for single-finger scanners:

- The most critical parameter is the acquisition area: reducing the area to the PIV IQS minimum requirement causes an average performance drop of 73% (see Fig. 2.23). A similar study, performed by NIST (Orandi et al., 2014), shows that cropping a probe fingerprint image from full size to FAP10 (corresponding to PIV IQS) and searching it over a gallery of full-size images leads to a relevant (~100%) FNIR drop. Further experiments are reported in Fernandez-Saavedra et al. (2016) where cropping is applied to fingerprints acquired with capacitive and thermal sensors.

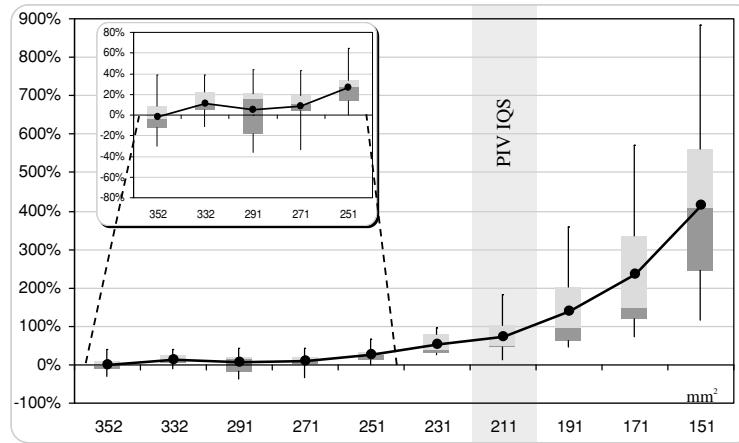


Fig. 2.23 Box-plot of the acquisition area experiment: each box graphically shows the descriptive statistics of the test carried out over the ten algorithms. The first five boxes are expanded in the inset graph to better show their statistics. The horizontal axis reports the minimum acquisition area requirements (in square millimeters) and the vertical axis shows the percentage performance drop. The minimum requirement based on the PIV IQS specification is highlighted (211 mm^2)

- Output resolution and geometric accuracy are also quite important: setting them to the PIV IQS minimum requirements leads to 20% and 1% performance drop, respectively.
- Other parameters, such as signal-to-noise ratio and gray range, do not seem to affect the automated recognition considerably: appreciable performance drop can be observed only for very strong degradations (well above PIV IQS requirements).

The simultaneous effect of all the parameters on recognition accuracy has only been briefly addressed in Cappelli et al. (2008), but it has been investigated in more detail in Alessandroni et al. (2008) with respect to single-finger scanners IQS adopted in certain recent large-scale applications (see Fig. 2.24). Their results show that.

- The simultaneous variation causes an average accuracy drop markedly higher than simply summing the individual accuracy drops. The total accuracy drop for PIV IQS is 156%.
- Enforcing “strong” constraints for acquisition area, output resolution, and geometric accuracy and “weak” constraints for the rest of the parameters is sufficient to assure good accuracy (accuracy drop of only 18%).

As mentioned earlier, sensing area is the most important parameter of fingerprint sensors. Many different experimental results on sensing area versus accuracy trade-off have been reported in the tests carried out by Jain et al. (1999), Maio et al. (2002), Schneider et al.

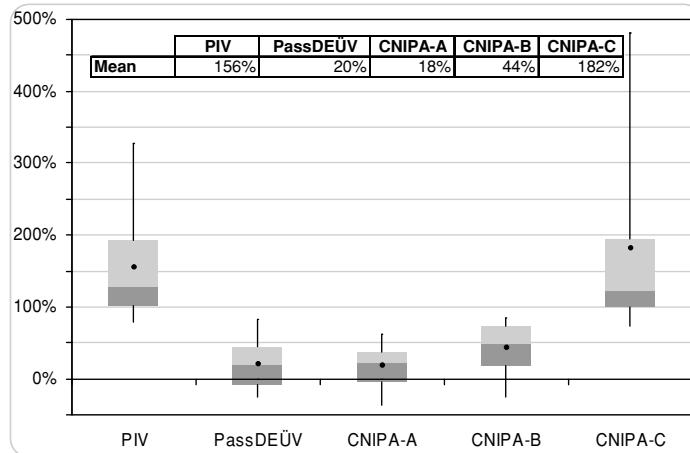


Fig. 2.24 Total accuracy drop for five different IQS specifications. PassDEÜV IQS was defined by German BSI (Federal Office for Information Security) for the German passport project and coincides with IAFIS IQS except for the area that can be smaller ($20 \times 16 \text{ mm}^2$). CNIPA-A/B/C specifications are being suggested by CNIPA (the Italian National Center for Information and Communication Technology in the Public Administration, now called AGID)

(2003), Ross and Jain (2004), Marcialis and Roli (2004), BioLab (2007), IDTL (2018), and AlShehri et al. (2018). In all of these studies, the authors compared the accuracy of one or more fingerprint recognition algorithms on two fingerprint databases acquired under the same conditions except the scanner. The first database is acquired using a large-area (usually optical) scanner, and the second database is acquired using a small-area (usually solid state) scanner. Table 2.3 summarizes the results of these studies and points out the significant accuracy drop observed in each one of them. Although the reported drop in accuracy is largely due to the area reduction, the sensing area is not the only parameter that varies between the two scanners used in the experiments. Hence, this accuracy drop is a consequence of a mix of parameter changes. It is worth noting that the accuracy drop reported here is generally higher than that plotted in Fig. 2.23, where the impact of the sensing area has been isolated from the other parameters. In conclusion, from Table 2.3 studies we can expect an accuracy drop ranging from $10 \times$ to $100 \times$ when using very small-area sensors (<<FAP10) instead of large-area ones (FAP30).

2.8 Examples of Fingerprint Scanners

Several fingerprint scanners based on the various sensing technologies surveyed in Sect. 2.3 are commercially available. Certainly, the main characteristics of a fingerprint scanner depend on the specific sensor used in it, which in turn determines the image

Table 2.3 Performance drop in terms of fingerprint verification accuracy obtained on two datasets collected with two different scanners, where the second scanner area is substantially lower than the first scanner area. Scanning area is reported as width (w) \times height (h). In Maio et al. (2002) and BioLab (2007), the accuracy drop is the average drop over the three best-performing matching algorithms. For IDTL (2018), we report accuracy drop on the metric FNMR@FMR = 0.001% (Table 5 in the paper). For AlShehri et al. (2018), the accuracy drop is the average drop over three matching approaches (Tables 4–6 in the paper)

Experiment	First scanner	Second scanner	Performance drop (%)
Jain et al. (1999)	1.0" \times 0.96" (500 dpi) Optical, FTIR	0.6" \times 0.6" (500 dpi) Capacitive	220
Maio et al. (2002)–FVC2002	0.78" \times 0.75" (500 dpi) Optical, FTIR	0.6" \times 0.6" (500 dpi) Capacitive	222
Schneider et al. (2003)	0.8" \times 1.2" (500 dpi) Ultrasound	0.38" \times 0.38" (500 dpi) Ultrasound	1200
Ross and Jain (2004)	1.0" \times 0.96" (500 dpi) Optical, FTIR	0.6" \times 0.6" (500 dpi) Capacitive	277
Marcialis and Roli (2004)	0.52" \times 0.98" (569 dpi) Optical, FTIR	0.5" \times 0.5" (500 dpi) Capacitive	544
BioLab (2007)–FVC2006	0.7" \times 0.98" (569 dpi) Optical, FTIR	0.55" \times 0.02" (500 dpi) Thermal (swipe)	3136
BioLab (2007)–FVC2006	0.7" \times 0.98" (569 dpi) Optical, FTIR	0.38" \times 0.38" (250 dpi) Capacitive (Active)	11,832
IDTL (2018)	0.47" \times 0.67" (385 dpi) Thermal (Active)	0.38" \times 0.38" (508 dpi) Capacitive	282
IDTL (2018)	0.47" \times 0.67" (385 dpi) Thermal (Active)	0.30" \times 0.31" (373 dpi) Capacitive	778
AlShehri et al. (2018)	1.28" \times 0.96" (500 dpi) Optical, FTIR	0.42" \times 0.55" (363 dpi) Capacitive	2593
AlShehri et al. (2018)	1.28" \times 0.96" (500 dpi) Optical, FTIR	0.29" \times 0.29" (500 dpi) Capacitive (Active)	5023

Table 2.4 Some examples of multi-finger scanners. FAP45 scanners are for two fingers or rolled fingerprint acquisition. FAP50 and FAP60 can acquire up to four fingers simultaneously. Note that not also non-optical-FTIR scanners have been certified by the FBI under Appendix F IQS. Companies are listed in alphabetical order

Company	Model	Technology	Dpi	Area ($w \times h$)	IAFIS IQS compliant
HID Global www.hidglobal.com	HID Guardian 200	Optical FTIR	500	3.2" × 3.3" (FAP 60)	✓
HID Global www.hidglobal.com	Nomad 60	Capacitive TFT	500	3.0" × 3.0" (FAP 60)	✓
Integrated Biometrics www.integratedbiometrics.com	Five-O	Optical electro-luminescent	500	3.2" × 2.0" (FAP 50)	✓
Papillon www.papillon.ru	DS-30NM	Optical FTIR	500	3.38" × 3.07" (FAP 60)	✓
Thales www.thalesgroup.com	DactyScan40i	Optical FTIR	500	1.6" × 1.6" (FAP 45)	✓

parameters introduced in Sect. 2.5 as well as other characteristics such as size, cost, and durability. However, many other features should also be taken into account when choosing a fingerprint scanner for a specific application.

- *I/O Interface*: almost all fingerprint scanners have digital output, and they directly interface to an external computer through a USB interface. For scanner modules to be integrated into embedded systems, an SPI interface is typically available.
- *Frames per second*: it indicates the number of images a touch scanner is able to acquire and send to the host in 1 s. A high frame rate (e.g., larger than 5 frames/s) better tolerates movements of the finger on a touch scanner and allows a more friendly interaction with the scanner. It can also provide a natural visual feedback during the acquisition.
- *Automatic finger detection*: some scanners automatically detect the presence of a finger on the acquisition surface, without requiring the host to continually grab and process frames; this allows the acquisition process to be automatically initiated as soon as the user's finger touches the sensor.
- *Encryption*: securing the communication channel between the scanner and the host is an effective way of securing a system against attacks (see Chap. 9). For this purpose, some commercial scanners implement state-of-the-art symmetric and public-key encryption capability.
- *Hardware support for Presentation Attack Detections*: some devices integrate specific hardware components (e.g., a heart rate monitor), to support presentation attack detection (see Chap. 9).

Table 2.5 Some examples of single-finger scanners. Companies are listed in alphabetical order. The FAP level reported in the Area column only refers to the sensing area and does not consider the PIV IQS compliance, which is separately reported in the last column

Company	Model	Technology	Dpi	Area (w × h)	PIV IQS
Biometrika www.biometrika.it	HiScanPro	Optical FTIR	500	1" × 1" (FAP 30)	✓
HID Global www.hidglobal.com	Nomad 30	Capacitive TFT	500	0.8" × 1.0" (FAP 30)	✓
HID Global www.hidglobal.com	DigitalPersona 4500	Optical FTIR	512	0.57" × 0.71" (FAP 10)	
HID Global www.hidglobal.com	Lumidigm V-Series	Optical Multispectral	500	0.7" × 1.11" (FAP 20)	
IDEAMIA www.idemia.com	MSO350	Optical FTIR	500	0.86" × 0.86" (FAP 20)	✓
Next Biometrics www.nextbiometrics.com	Access 100Pro	Thermal Active	385	0.46" × 0.66"	
Secugen www.secugen.com	Hamster Pro20	Optical FTIR	500	0.6" × 0.9" (FAP 20)	✓

- *Supported operating systems:* depending on the application and the infrastructure where the fingerprint scanners have to be employed, compatibility with several operating systems, and in particular the support of open-source operating systems such as Linux, could be an important feature.

Tables 2.4 and 2.5 list some multi-finger and single-finger scanners, respectively. In Table 2.6, we report a list of vendors providing sensors/modules for mobile phone or payment card integration. In general, the cost of fingerprint scanners decreased significantly with mass adoption in many applications. In 2020, the cost of a multi-finger scanner is typically > US\$1,500; the cost of single-finger large-area scanners varies, usually in the range US\$50–US\$500, according to the size of the acquisition area and the quality of the image produced. Cheaper models are generally used for personal or corporate applications such as logon to a PC or a network; high-quality and large-area (\geq FAP30) models are used in large-scale applications such as border-crossing and e-passports. Finally, the cost of small-area sensors, to be integrated into mobile devices, is about US\$2.

Figure 2.25 shows an image acquired with a multi-finger scanner and the result of automatic slap segmentation. Figure 2.26 shows fingerprint images of the same finger acquired with different single-finger scanners.

Fingerprint scanners often fail in producing good-quality images when the finger skin condition is overly dry or overly wet (see Fig. 2.27), and fingerprint recognition accuracy degrades accordingly (Krishnasamy et al., 2011). Certain sensing technologies (or certain special coatings applied to the sensor surface) seem to better tolerate certain types of skin conditions. A few studies on the scanner robustness with respect to suboptimal skin conditions can be found in Kang et al. (2003) and Yau et al. (2004). However, systematically

Table 2.6 Some examples of sensors and modules specifically designed for mobile devices, payments cards, and/or portable objects with curved surfaces. Technical details, such as sensing area and resolution, are often non-publicly available for this category of sensors

Company	Product	Technology	Notes
Apple www.apple.com		Capacitive Active	First successful integration of a sensor in a mobile phone: iPhone5S (2013)
Fingerprint Cards www.fingerprints.com	FPC series	Capacitive	Since 2014, it produces small sensors to be mounted in the home button or backside of mobile phones
Goodix www.secugen.com		Optical TFT	In-display integration in Huawei P30 (2019)
Idx www.idexbiometrics.com	IDX series	Capacitive	Specifically designed for payment cards
Isorg www.isorg.fr		Optical TFT	Flexible plastic substrate, suitable for curved surfaces
Qualcomm www.qualcomm.com	3D Sonic	Ultrasound	In-display integration in Samsung Galaxy S10 (2019)
Synaptics www.synaptics.com	Clear ID	Optical TFT	In-display integration in Xiaomi Mi8 (2018)
TouchBiometrix www.touchbiometrix.com		Capacitive TFT	Flexible plastic substrate, suitable for curved surfaces

studying the device robustness to skin conditions would require presenting fingers with an identical skin condition, pressure, etc., to different devices. To this purpose, Engelsma et al. (2018) proposed a molding and casting process that can create three-dimensional fingerprint targets with the mechanical, optical, and electrical properties of human skin. They argued that mounting such universal targets on a robotic hand would allow to better quantify and compare device robustness and interoperability. Previous attempts to manufacture three-dimensional fingerprints were reported by Arora et al. (2016, 2017).

Finally, inherently poor-quality fingers, whose ridges are spoiled and damaged by scars and creases, typically produce low-quality fingerprint images with most of today's mature sensing technologies (see Fig. 2.28). The last generation of ultrasound and OCT scanners, being able to get information from under the surface, in principle are more robust with respect to surface damages (see Fig. 2.15). However, no systematic and vendor-independent study is still available to support this argument.



Fig. 2.25 An example of 500 dpi image (1558×1691 pixels) acquired with the multi-finger scanner Papillon DS-30; the four rectangles show the position of the four fingerprints as located by an automatic slap segmentation algorithm

2.9 Dealing with Small-Area Sensors

As shown in Sect. 2.7, recognizing fingerprints acquired through sensors that capture only a partial fingerprint is difficult due to the possibility of having only a small overlap between different acquisitions of the same finger. For a minutiae-based matching algorithm (see Sect. 4.3), a small overlap between two fingerprints leads to a small number of minutiae correspondences and thereby to lower match confidence (Yau et al., 2000; Pankanti et al. 2002; Jain & Ross, 2002). This effect is even more marked on intrinsically poor-quality fingers, where only a subset of the minutiae can be reliably extracted

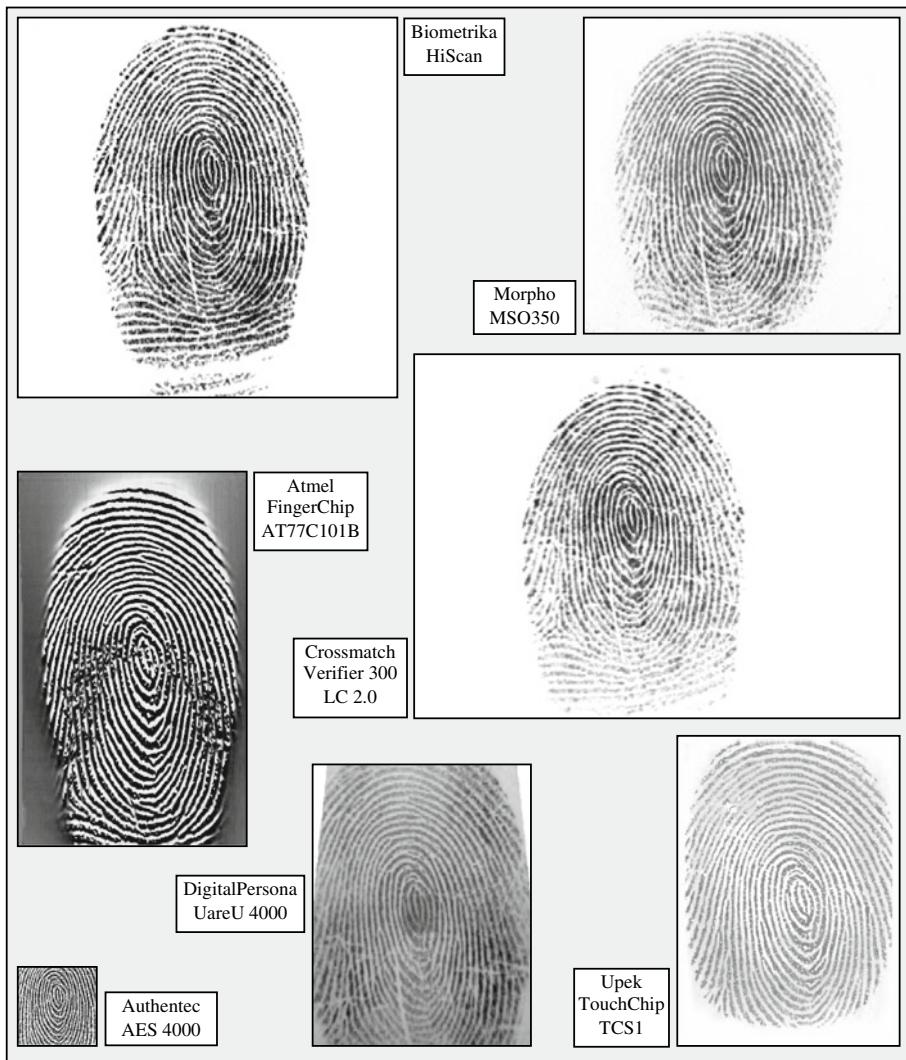


Fig. 2.26 Fingerprint images of the same finger acquired by single-finger scanners of different technologies. Images are shown with true proportions

and used. A small overlap also affects the reliability of both correlation-based and non-minutiae feature-based matching techniques, since the amount of information available for the matching is reduced.

Roy et al. (2017) pointed out the vulnerability of partial fingerprint-based authentication by demonstrating the feasibility of generating “MasterPrints” (at minutiae level)

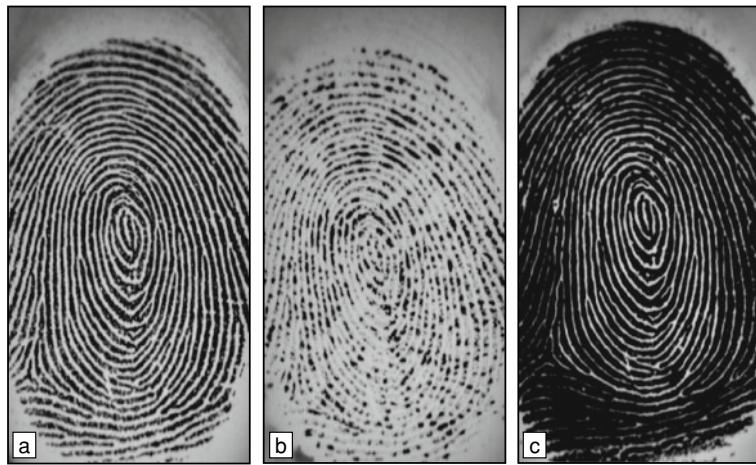


Fig. 2.27 Three fingerprint images of the same finger with different skin conditions acquired with an optical FTIR scanner: **a** normal, **b** dry, and **c** wet

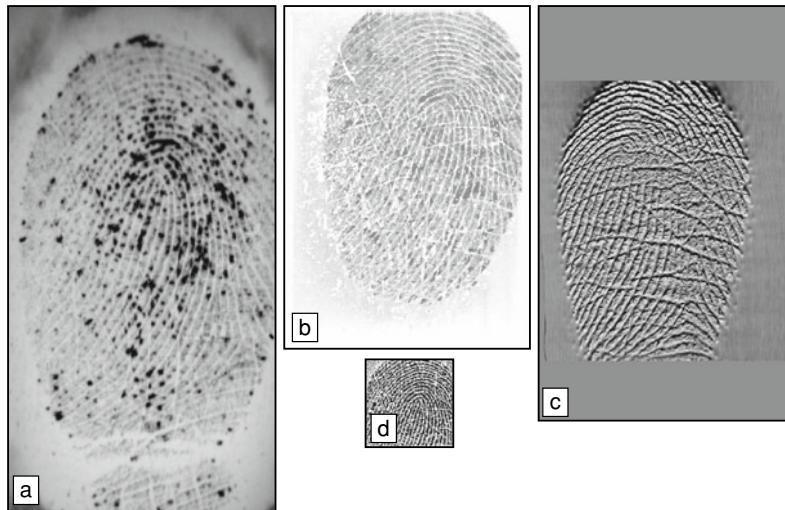


Fig. 2.28 Fingerprint images of the same poor-quality finger acquired with scanners based on four different sensing technologies: **a** Optical FTIR, **b** Capacitive, **c** Thermal, and **d** Capacitive (Active)

that have high probability to be matched with small fingerprint fragments. In their experiments with a capacitive sensor, similar to the one used by iPhone5S, they were able to spoof the system 6.60% of the time in a single attempt when FMR was set to 0.1% and each subject was enrolled with one finger and 12 partial impressions per finger. Bontrager

et al. (2018) showed that Generative Adversarial Networks (GAN) can be used to generate Deep-MasterPrints at the image level, thus making a presentation attack easier. In their study, Deep-MasterPrints were more effective than feature-level MasterPrints, being able to successfully spoof the system 22.5% of the time at $FMR = 0.1\%$.

The most common approach to mitigate the accuracy drop induced by small sensors is to create a full-size fingerprint representation during enrollment by fusing multiple fragments. This is nowadays a largely diffused procedure during fingerprint enrollment in mobile devices, where a user-friendly graphical animation on the display supports the user in placing his/her finger several times (about 25 touches according to Mathur et al., 2016). This approach, known as fingerprint *mosaicking*, is sketched in Fig. 2.29. At recognition time, the user touches the sensor once, and the small fingerprint area captured is matched against the large reference fingerprint. Although the area of overlap is increased through mosaicking, the fingerprint image acquired at the recognition time is still small that usually leads to lower recognition accuracy as compared to the large-area sensors.

Different approaches could be adopted to combine multiple (partially overlapping) fingerprint images into a large fingerprint image (Brown, 1992). A simple idea is to exploit the same algorithm used for matching: in fact, a byproduct of matching two fingerprints is an estimate of the transformation (e.g., displacement and rotation) between the two fingerprints; given this information, fusing more images into a single large image can be accomplished by superimposing the aligned images and by appropriately weighting the intensity of the corresponding pixels. The main difficulty in creating a mosaicked image is that, often, the alignment between the various impressions/pieces cannot be completely recovered due to non-linear skin distortion. For example, in Fig. 2.29, the ridge pattern is blurred in the overlapping area.

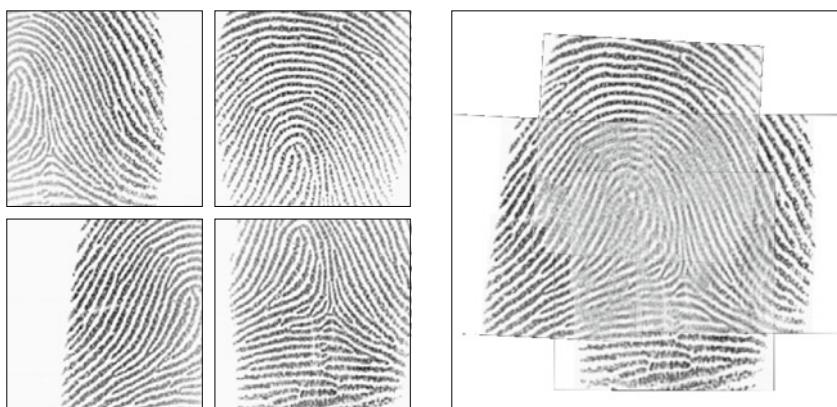


Fig. 2.29 A fingerprint mosaic image obtained by combining four fingerprint images acquired with a 0.51×0.51 square inch 500 dpi optical sensor

For some matching techniques (especially feature-based matching), the blurring and artifacts created by image mosaicking in the overlapping regions can be troublesome. Therefore, some authors (Mathur et al., 2016) prefer to keep isolated the partial fingerprints acquired during enrollment and to perform multiple partial-to-partial comparisons at verification time.

The mosaicking algorithm of Jain and Ross (2002) combines two impressions of a finger by first aligning the fingerprint images by using the Iterative Closest Point (ICP) algorithm (Besl & McKay, 1992). A low-pass filter is initially applied to the images and the pixel intensities are sub-sampled to a narrow gray-scale range of [10, 20] to ensure a fairly smooth change in the surface corresponding to the range image of the fingerprints (see Fig. 3.9 for an example of surface representation of a fingerprint image). The images are then segmented into foreground and background, an alignment between the two foreground images is estimated using the ICP algorithm, and a composite image is created by taking an average of the corresponding pixel intensities (see Fig. 2.30). Jain and Ross (2002) compared the mosaicking of fingerprints at the gray-scale image representation level with mosaicking at the minutiae representation level and found the former to outperform the latter. Their results on 320 query images showed an improvement of ~4% in the fingerprint verification accuracy when two impressions of each finger were combined to form a composite template at the gray-scale intensity level.

To compensate for the amount of plastic distortion between two partial images, Lee et al. (2003) and Choi et al. (2007) use non-rigid alignment transformations such as Chamfer matching (Borgefors, 1988) and the Thin Plate Spline (TPS) model (Ross et al., 2006a). The transform is initially estimated with matched minutiae and then refined by matching ridges. In Choi et al. (2007), unpaired ridges in the overlapping area between two images are iteratively matched by minimizing the registration error, which consists

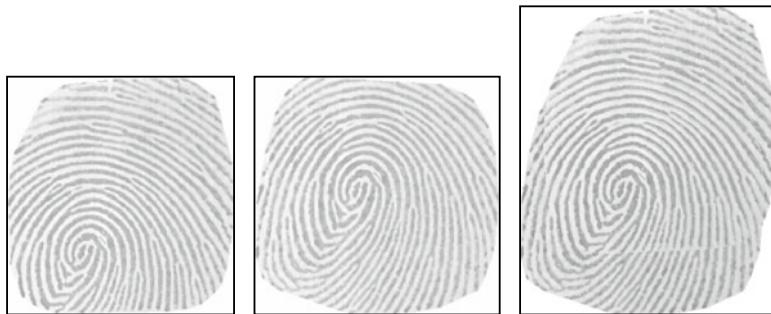


Fig. 2.30 Combination of two impressions of a finger (on the left) into one composite fingerprint image (on the right) using the Jain and Ross (2002) method. © IEEE. Reprinted, with permission, from Jain and Ross (2002)

of the ridge matching error and the inverse consistency error. During the estimation, erroneous correspondences are eliminated by considering the geometric relationship between the correspondences and by checking whether or not the registration error is minimized.

Wang et al. (2018) proposed a mosaicking scheme where (i) an optimal central patch is initially selected, (ii) the remaining patches are coarsely aligned with the central patch, and (iii) the final positions of the patches are globally optimized. Dense registration techniques (introduced in Sect. 4.5.4) can be very accurate for fingerprint mosaicking, but their computational complexity is generally high.

Other image mosaicking techniques were proposed by Ratha et al. (1998), Zhou et al. (2001), Know et al. (2010), and Zhang et al. (2013), with the aim of obtaining a rolled fingerprint impression from a sequence of flat fingerprint images acquired through a live-scan scanner. Ratha et al. (1998) mosaicking algorithm consists of the following steps: (i) segment fingerprint foreground and background areas in each fingerprint image, (ii) use the foreground mask to weight each image's contribution, (iii) stack the weighted gray-scale frames to compute the mosaicked gray-scale image, and (iv) stack the foreground masks to compute a confidence index at each pixel. Because all the fingerprint images to be combined are acquired in a single rolling of a fingerprint, the authors assume that the images are already aligned. Furthermore, the foreground masks are shrunk so that only the central portion of each image with the best contrast and least distortion is used. The method proposed by Know et al. (2010) finds dense correspondences between images and warps the entire fingerprint area by using a registration method based upon a Markov random field (MRF) energy model.

For minutiae-based matching algorithms, instead of rebuilding the whole fingerprint image, it may be more convenient to build the mosaic at the feature level; that is, rearranging the minutiae from each fingerprint impression into a single map. Practical implementations of this approach, known in the literature as *template consolidation* or *template improvement*, are discussed in Yau et al. (2000), Toh et al. (2001), Jain and Ross (2002), Jiang and Ser (2002), Ramoser et al. (2002), Yeung et al. (2004), Ryu et al. (2005, 2006), Sha et al. (2007), Uz et al. (2009), and Hu et al. (2017). In the experiment carried out by Ross et al. (2006b), feature-level mosaicking outperforms image-based mosaicking. Finally, in Yang and Zhou (2006), different schemes to combine multiple enrolled impressions are comparatively studied. Their experimental results show that a larger improvement can be obtained by using a decision fusion scheme rather than feature fusion. However, the experiments have been carried out on databases collected with medium to large-area sensors, and it is not clear if they are valid for small-area sensors as well.

2.10 Storing and Compressing Fingerprint Images

In many law enforcement and government applications of AFIS, the size of the fingerprint database is large. For example, Hopper et al. (1993) estimated the size of the FBI fingerprint card archive (each card is a paper form containing all ten ink fingerprints from two hands of an individual along with his demographic information) to be over 200 million, occupying an acre of filing cabinets in the J. Edgar Hoover building in Washington, DC. Furthermore, the archive size was increasing at the rate of 30,000–50,000 new cards per day. The digitization of fingerprint cards was an obvious choice, although the size of the resulting digital archive was also problematic. Each fingerprint impression, when digitized at 500 dpi, produces an image with 768×768 pixels at 256 Gray levels. An uncompressed representation of such an image requires 589,824 bytes and about 10 megabytes are necessary to encode a single card (both a dab and rolled impression of each finger are present on the card). A simple multiplication by 200 million yields the massive storage capacity of 2,000 terabytes for the entire archive. Another problem was the long delay involved in transmitting a fingerprint image over a band-limited communication channel: about 3 h were needed for transmitting a single image over a 9,600 baud channel.

Today, the availability of large storage systems (i.e., with petabytes capacity) and fast communication channels (i.e., 5G) makes fingerprint image storage and transmission less critical. Nevertheless:

- Fingerprint images need also to be stored inside the chip of electronic documents (e.g., in biometric passports) where only a few tens of KB are available.
- The cost of large-size storage (including its backup) can be an obstacle to the diffusion of civil AFIS in emerging countries.

Therefore, the need for an effective compression technique is still very important. Unfortunately, neither the well-known lossless methods nor the JPEG compression method was satisfactory. The former typically provides a compression ratio of 2 when applied to gray-scale fingerprint images and the latter, at the FBI target compression ratio (0.75 bit per pixel, i.e., about 1:10.7), produces block artifacts due to the independent compression through Discrete Cosine Transform (DCT) of single 8×8 image blocks (see Fig. 2.31c).

A compression technique (with small, acceptable loss), called Wavelet Scalar Quantization (WSQ), was developed on the basis of the work by Hopper and Preston (1991), Bradley et al. (1992), Hopper et al. (1993), and Brislaw et al. (1996). Due to its superiority with respect to other general-purpose compression techniques, WSQ became the FBI standard for the compression of 500 dpi fingerprint images (CJIS, 2010). WSQ is based on adaptive scalar quantization of a discrete wavelet transform (Hopper et al., 1993; Onyshczak & Youssef, 2004). The WSQ encoder performs the following steps:

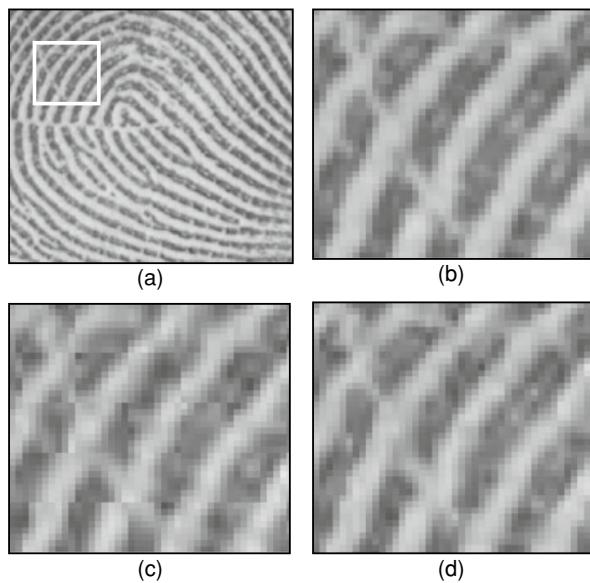


Fig. 2.31 Fingerprint compression: **a** the central section of a fingerprint image scanned at 500 dpi resolution; **b** the marked portion of the image in **a**; **c** the marked portion of the image in **a** after the image was compressed using a generic JPEG (www.jpeg.org) image compression algorithm; and **d** the marked portion of the image in **a** is shown after the image was compressed using the WSQ compression algorithm. Both JPEG and WSQ examples used a compression ratio of 1:12.9; JPEG typically introduces blocky artifacts and obliterates detailed information. Image courtesy of Chris Brislawn, Los Alamos National Laboratory

1. Fingerprint image is decomposed into a number of spatial frequency sub-bands (typically 64) using a Discrete Wavelet Transform (DWT).
2. Resulting DWT coefficients are quantized into discrete values; this is the step that leads to the loss of information and makes it very difficult to invert the process to obtain the exact starting image.
3. Quantized sub-bands are concatenated into several blocks (typically three to eight) and compressed using adaptive entropy encoding (Huffman run-length).

A compressed image can be decoded into the original image (with certain loss) by applying the equivalents of the above steps in reverse order (i.e., Huffman decoding, quantization decoding, and Inverse Discrete Wavelet Transform [IDWT]).

WSQ can compress a fingerprint image by a factor of 10–25 (see Fig. 2.31). A typical compression ratio of 10–15 seems to be most appropriate, as higher compression ratios result in an unacceptable degradation of the fingerprint image.

Although the common DCT-based JPEG compression is unsatisfactory for highly compressed fingerprint images, an evolution of the same general-purpose image compression,

known as JPEG 2000 (Skodras et al., 2001), can actually compete with WSQ. JPEG 2000 image coding is detailed in the standard ISO/IEC 15444-1 (2019). Analogous to WSQ, JPEG 2000 is also based on DWT. Since DWT is applied to the whole image and not to separate image blocks, JPEG 2000 compressed images are no longer affected by the undesired blocky artifacts. The differences between WSQ and JPEG 2000 are in the methods used for the decomposition, quantization, and entropy encoding (Allinson et al., 2007).

Based on the experimentations carried out until now over different sources and for different compression rates (Lepley, 2001; Figueroa-Villanueva et al., 2003; Allinson et al., 2007), it seems that JPEG 2000 is slightly better than WSQ for typical compression rates (e.g., 1:15) and noticeably better for high compression rates (e.g., 1:32). A more recent study explicitly focusing on 500 dpi resolution (Libert et al., 2012) pointed out that WSQ seems to better retain lower frequency components (i.e., fingerprint ridge structure and minutiae) at the cost of degrading more rapidly at high frequencies (e.g., small ridge details); JPEG 2000, on the other hand, exhibits a more linear degradation across the frequency bands with less loss at high frequencies.

Both WSQ and JPEG 2000 compressions are now allowed by fingerprint standards. ANSI/NIST-ITL 1-2011 (2015) recommends (i) WSQ for 500 dpi images with compression limited to 1:10 for PIV IQS compliant devices and to 1:15 for IAFIS IQS compliant devices; (ii) JPEG 2000 (max. compression 1:10) for 1000 dpi resolutions.

Some improvements and efficient implementations of the basic WSQ approaches have been proposed (Kasaei et al., 1997, 2002; Deriche et al., 1999; Eslami & Radha, 2004; Zhao & Wang, 2009). Alternative approaches have been investigated where the compression relies on explicitly extracting features from fingerprints instead of encoding raw pictorial information (Abdelmalek et al., 1984; Chong et al., 1992). For example, in Chong et al. (1992), after binarization and thinning of the fingerprint image (see Sect. 3.7.1), the one-pixel-wide ridges are encoded by B-spline curves. This allows compression ratios of 1:20–1:25 to be achieved, but since only a feature-based representation of the image is used, the gray-scale fingerprint image cannot be reconstructed.

2.11 Summary

In the last two decades, there was a remarkable trend toward reducing both the cost and the size of the scanners to expand the market of personal and corporate applications. This trend further accelerated in 2013 when Apple launched iPhone5S with fingerprint sensors onboard, starting the era of mobile biometric authentication.

Several new companies were founded and new fingerprint image acquisition technologies were developed; some manufacturers went too far and reduced the sensor size and image quality to such an extent that recognition accuracy suffered significantly. Swipe sensors, whose usability was never improved enough to allow effortless usage, are now being abandoned. In the years 2013–2018, small-area capacitive sensors have been the

default choice for mobile phone integration, even if their vulnerability was pointed out in a number of studies.

Fingerprint-based authentication is now more mature. Image quality specifications for fingerprint scanners have been introduced to certify the scanners for law enforcement and government applications. Several large-scale applications have been successfully deployed (including the Indian Aadhaar project) and system integrators are more knowledgeable of the role of scanner quality in determining the overall system accuracy. Consequently, some of the acquisition technologies have been abandoned and the manufacturers have concentrated on those that offer the best performance/cost trade-off for the specific application. However, existing image quality specifications (IQS) are still not perfect and do not cover all the relevant aspects (e.g., the ability of a scanner to acquire dry or wet fingers is not covered by any IQS), so they are “necessary but not sufficient”.

For AFIS and large-scale civilian applications where large-area scanners (both multi-finger and single-finger) are required, the traditional optical-FTIR scanners are being challenged by new, flat portable devices (based on TFT backplane) that were recently certified by the FBI.

Mid-size area scanners based on different technologies are dominating the market of logical security applications such as logon to a PC or to a network as well as employee physical access control and attendance. Smartphone scanners are also rapidly changing, abandoning small capacitive sensors integrated into the home bottom in favor of optical or ultrasound in-display sensors. Here too, the possibility of reducing the cost due to the TFT process will yield an increased sensing area, allowing more fingers to be acquired simultaneously.

At the same time, fingerprint sensors on transparent, flexible backgrounds are being introduced, and we can envision a new generation of portable devices with a fingerprint sensing surface. Payment cards with embedded fingerprint sensors could be one of the next mass applications of fingerprint authentication. Automotive is another sector where biometric authentication devices have been announced (and prototyped) for a long time, but never reached maturity. In fact, still today, no fingerprint sensors have proved resistant enough to meet the demanding automotive requirements (extended temperature ranges, mechanical shocks, etc.), especially for outdoor authentication. Furthermore, while smartphone unlock or pay-by-touch for small value transactions are usually not considered as security-critical applications, vehicles are much more expensive than mobile phones and engine ignition needs to be properly secured (low False Match Rate and effective Presentation Attack Detection) to avoid car thefts.

References

- Abdelmalek, N., Kasvand, T., Goupil, D., & Otsu, N. (1984). Fingerprint data compression. In *Proceedings of International Conference on Pattern Recognition* (7th ed., pp. 834–836).
- Alessandroni, A., Cappelli, R., Ferrara, M., & Maltoni, D. (2008). Definition of fingerprint scanner image quality specifications by operational quality. In *Proceedings of European Workshop on Biometrics and Identity Management*.
- Allinson, N. M., Sivarajah, J., Gledhill, I., Carling, M., & Allinson, L. J. (2007). Robust wireless transmission of compressed latent fingerprint images. *IEEE Transactions on Information Forensics and Security*, 2(3), 331–340.
- AlShehri, H., Hussain, M., AboAlSamh, H., & AlZuair, M. (2018). A large-scale study of fingerprint matching systems for sensor interoperability problem. *Sensors*, 18(4), 1008.
- ANSI/NIST-ITL 1-2011. (2015). NIST, *Data format for the interchange of fingerprint, facial & other biometric information*, update 2015 of NIST Special Publication 500-290e3.
- Arora, S. S., Cao, K., Jain, A. K., & Paultre, N. G. (2016). Design and fabrication of 3D fingerprint targets. *IEEE Transactions on Information Forensics and Security*, 11(10), 2284–2297.
- Arora, S. S., Jain, A. K., & Paultre, N. G. (2017). Gold fingers: 3D targets for evaluating capacitive readers. *IEEE Transactions on Information Forensics and Security*, 12(9), 2067–2077.
- Auksorius, E., & Boccara, A. C. (2015). Fingerprint imaging from the inside of a finger with full-field optical coherence tomography. *Biomedical Optics Express*, 6(11), 4465–4471.
- Auksorius, E., & Boccara, A. C. (2017). Fast subsurface fingerprint imaging with full-field optical coherence tomography system equipped with a silicon camera. *Journal of Biomedical Optics*, 22(9), 1–8.
- Auksorius, E., Raja, K. B., Topcu, B., Ramachandra, R., Busch, C., & Boccara, C. A. (2020). Compact and mobile full-field optical coherence tomography sensor for subsurface fingerprint imaging. *IEEE Access*, 8, 15194–15204.
- Aum, J., Kim, J., & Jeong, J. (2016). Live acquisition of internal fingerprint with automated detection of subsurface layers using OCT. *IEEE Photonics Technology Letters*, 28(2), 163–166.
- Bae, S., Ling, Y., Lin, W., & Zhu, H. (2018). Optical fingerprint sensor based on a-Si:H TFT technology. *Proceedings of SID Symposium Digest of Technical Papers*, 49(1), 1017–1020.
- Besl, P. J., & McKay, N. D. (1992). A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2), 239–256.
- Bicz, W., Banasiak, D., Bruciak, P., Gumienny, S., Gumuliński, Z., Kosz, D., Krysiak, A., Kuczyński, W., Pluta, M., & Rabiej, G. (1999). Fingerprint structure imaging based on an ultrasound camera. *Instrumentation Science and Technology*, 27, 295–303.
- BioLab. (2007). BioLab—University of Bologna, *FVC 2006 web site*. Retrieved November 27, 2008, from <http://bias.csr.unibo.it/fvc2006>.
- Bontrager, P., Roy, A., Togelius, J., Memon, N., & Ross, A. (2018). DeepMasterPrints: Generating masterprints for dictionary attacks via latent variable evolution. In *Proceedings of International Conference on Biometrics Theory, Applications and Systems (BTAS)*, Redondo Beach, CA, USA (pp. 1–9).
- Borgefors, G. (1988). Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(8), 849–865.
- Bradley, J. N., Brislaw, C. M., & Hopper, T. (1992). The FBI wavelet/scalar quantization standard for grayscale fingerprint image compression. In *Proceedings of SPIE (Visual Info. Proc. II)* (pp. 293–304).

- Brislawn, C. M., Bradley, J. N., Onyshczak, R. J., & Hopper, T. (1996). The FBI compression standard for digitized fingerprint images. In *Proceedings of SPIE (Applications of Digital Image Processing XIX)* (Vol. 2847).
- Brown, L. G. (1992). Image registration techniques. *ACM Computing Surveys*, 24(4), 326–376.
- Cappelli, R., Ferrara, M., & Maltoni, D. (2008). On the operational quality of fingerprint scanners. *IEEE Transactions on Information Forensics and Security*, 3(2), 192–202.
- Cappelli, R., Maio, D., Maltoni, D., Wayman, J. L., & Jain, A. K. (2006). Performance evaluation of fingerprint verification systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1), 3–18.
- Champod, C., Lennard, C. J., Margot, P., & Stoilovic, M. (2016). *Fingerprints and other ridge skin impressions* (2nd ed.). CRC Press.
- Chen, Y., Parziale, G., Diaz-Santana, E., & Jain, A. K. (2006). 3D touchless fingerprints: Compatibility with legacy rolled images. In *Proceedings of Biometric Symposium*.
- Choi, H., Choi, K., & Kim, J. (2010). Mosaicing touchless and mirror-reflected fingerprint images. *IEEE Transactions on Information Forensics and Security*, 5(1), 52–61.
- Choi, K., Choi, H., Lee, S., & Kim, J. (2007). Fingerprint image mosaicking by recursive ridge mapping. *IEEE Transaction on Systems, Man, and Cybernetics, Part B*, 37(5), 1191–1203.
- Chong, M., Gay, R., Tan, H., & Liu, J. (1992). Automatic representation of fingerprints for data compression by B-spline function. *Pattern Recognition*, 25(10), 1199–1210.
- Chugh, T., & Jain, A. K. (2019). *OCT fingerprints: Resilience to presentation attacks*. arXiv:1908.00102.
- CJIS. FBI—CJIS Division. (2005). *Electronic fingerprint transmission specification (EFTS)*. Int. Report: IAFIS-DOC-01078-7.1 (V7.1).
- CJIS. FBI—CJIS Division. (2006). *Image quality specifications for single finger capture devices*. Retrieved July, 2021, from <https://fbibiospecs.fbi.gov/file-repository/pivspec.pdf/view>.
- CJIS. FBI—CJIS Division. (2010). *WSQ gray-scale fingerprint image compression specification—Version 3.1*. Retrieved March, 2022, from https://fbibiospecs.fbi.gov/file-repository/wsqt_gray-scale_specification_version_3_1_final.pdf/view.
- CJIS. FBI—CJIS Division. (2017). *Electronic biometric transmission specification (EBTS)*. Int. Report: NGI-DOC-01862-1.1 (V10.0.8). Retrieved March, 2022, from https://fbibiospecs.fbi.gov/file-repository/master-ebts-v10-0-8-09302017_final.pdf.
- Clausen, S. (2007). A single-line AC capacitive fingerprint swipe sensor. In N. K. Ratha & V. Govindaraju (Eds.), *Advances in biometrics: Sensors, algorithms and systems* (pp. 49–62). Springer.
- Colins, M.W. (1992). *Realizing the full value of latent prints*. California Identification Digest.
- Darlow, L. N., & Connan, J. (2015). Efficient internal and surface fingerprint extraction and blending using optical coherence tomography. *Applied Optics*, 54(31), 9258–9268.
- Deriche, M., Kassaei, S., & Bouzerdoum, A. (1999). A novel fingerprint image compression technique using the wavelet transform and piecewise uniform pyramid lattice vector quantization. In *Proceedings of International Conference on Image Processing*.
- Donida Labati, R., Genovese, A., Piuri, V., & Scotti, F. (2016). Toward unconstrained fingerprint recognition: A fully touchless 3-D system based on two views on the move. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(2), 202–219.
- Edwards, D. G. (1984). *Fingerprint sensor*. US Patent 4429413.
- Engelsma, J. J., Arora, S. S., Jain, A. K., & Paulte, N. G. (2018). Universal 3D wearable fingerprint targets: Advancing fingerprint reader evaluations. *IEEE Transactions on Information Forensics and Security*, 13(6), 1564–1578.
- Engelsma, J. J., Cao, K., & Jain, A. K. (2019). RaspiReader: Open source fingerprint reader. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(10), 2511–2524.

- Eslami, R., & Radha, H. (2004). Wavelet-based contourlet transform and its application to image coding. In *Proceedings of International Conference on Image Processing* (Vol. 5, pp. 3189–3192).
- Fatehpuria, A., Lau, D. L., & Hassebrook, L. G. (2006). Acquiring a 2D rolled equivalent finger-print image from a non-contact 3D finger scan. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification III*.
- FBI. (2021). Retrieved March, 2022, from <https://fbibiospecs.fbi.gov/certifications-1/cpl>.
- Fernandez-Saavedra, B., Sanchez-Reillo, R., Ros-Gomez, R., & Liu-Jimenez, J. (2016). Small fingerprint scanners used in mobile devices: The impact on biometric performance. *IET Biometrics*, 5(1), 28–36.
- Ferrara, M., Franco, A., & Maltoni, D. (2007). Estimating image focusing in fingerprint scanners. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 30–34).
- Figueroa-Villanueva, M. A., Ratha, N. K., & Bolle, R. M. (2003). A comparative performance analysis of JPEG 2000 vs. WSQ for fingerprint image compression. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (4th ed., pp. 385–392).
- Fiumara, G., Tabassi, E., Flanagan, P., Grantham, J., Ko, K., Marshall, K., Schwarz, M., Woodgate, B., & Boehnen, C. (2017). *Nail to nail fingerprint challenge*. NIST-IR 8210. Retrieved July, 2021, from <https://nvlpubs.nist.gov/nistpubs/ir/2018/NIST.IR.8210.pdf>.
- Forkert, R. D., Kearnan, G. T., Nill, N. B., & Topiwala, P. N. (1994). *Test procedures for verifying IAFIS scanner image quality requirements*. MITRE Technical Report: MP 94B0000039R1.
- Galbally, J., Bostrom, G., & Beslay, L. (2017). Full 3D touchless fingerprint recognition: Sensor, database and baseline performance. In *Proceedings of International Joint Conference on Biometrics (IJCB)* (pp. 225–233).
- Garris, M. D., & McCabe, R. M. (2000). *NIST special database 27, fingerprint minutiae from latent and matching tenprint images*. U.S. National Institute of Standards and Technology.
- Gupta, P., & Gupta, P. (2012). Slap fingerprint segmentation. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, Arlington, VA (pp. 189–194).
- Gupta, P., & Gupta, P. (2014). An efficient slap fingerprint segmentation and hand classification algorithm. *Neurocomputing*, 142, 464–477.
- Gupta, P., & Gupta, P. (2016). An accurate slap fingerprint based verification system. *Neurocomputing*, 188, 178–189.
- Habegger, A., Mueller, L., Goette, J., & Jacomet, M. (2012). A subpixel-based fingerprint reconstruction algorithm. In *Proceedings of International New Circuits and Systems Conference* (pp. 41–44).
- Han, H., & Koshimoto, Y. (2008). Characteristics of thermal-type fingerprint sensor. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification V*.
- Han, Y., Nam, J., Park, N., & Kim, H. (2006). Resolution and distortion compensation based on sensor evaluation for interoperable fingerprint recognition. In *Proceedings of International Joint Conference on Neural Networks* (pp. 692–698).
- Hase, M., & Shimisu, A. (1984). Entry method of fingerprint image using a prism. *Transactions of the Institute of Electronic and Communication Engineers of Japan*, J67-D, 627–628.
- Hashido, R., Suzuki, A., Iwata, A., Okamoto, T., Satoh, Y., & Inoue, M. (2003). A capacitive finger-print sensor chip using low-temperature poly-Si TFTs on a glass substrate and a novel and unique sensing method. *IEEE Journal of Solid-State Circuits*, 38(2), 274–280.
- Hiew, B. Y., Teoh, A. B. J., & Pang, Y. H. (2007). Touch-less fingerprint recognition system. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 24–29).
- Hopper, T., Brislaw, C., & Bradley, J. (1993, February). *WSQ grayscale fingerprint image compression specification*. Federal Bureau of Investigation.
- Hopper, T., & Preston, F. (1991). Compression of grey-scale fingerprint images. In *Proceedings of SPIE 2242, Wavelet Applications* (pp. 309–318).

- Hu, Z., Li, D., Isshiki, T., & Kunieda, H. (2017). Narrow fingerprint template synthesis by clustering minutiae descriptors. *IEICE Transactions on Information and Systems*, E100-D(6), 1290–1302.
- Huang, S., Huang, Y., Yeh, C., Sugiura, N., You, J., & Peng, C. (2015). Design and modeling of 1000ppi fingerprint sensor. In *Proceedings of IEEE Sensors Conference*.
- Hwang, H., Lee, H., Jang, B., Kim, H., Lee, T., & Chae, Y. (2017). A 500-dpi transparent on-glass capacitive fingerprint sensor. In *Proceedings of SID Symposium Digest of Technical Papers*.
- IDTL - Carlos III University of Madrid. (2018). *Multi-sensor performance evaluation*.
- Inglis, C., Manchanda, L., Comizzoli, R., Dickinson, A., Martin, E., Mandis, S., Silveman, P., Weber, G., Ackland, B., & O’Gorman, L. (1998). A robust, 1.8 V, 250 mW, direct contact 500 dpi fingerprint sensor. In *Proceedings of IEEE Solid-State Circuits Conference*.
- Integrated Biometrics. (2019). *LES film technology*. Retrieved July, 2021, from <https://integratedbiometrics.com/wp-content/uploads/2020/03/LES-Film-Technology-Whitepaper.pdf>.
- ISO/IEC 19794-4. (2011). *ISO/IEC 19794-4:2011—Biometric data interchange formats—Part 4: Finger image data*. ISO/IEC Standard.
- ISO/IEC 15444-1. (2019). *Information technology—JPEG 2000 image coding system—Part 1: Core coding system*. ISO/IEC Standard.
- Iula, A. (2019). Ultrasound systems for biometric recognition. *Sensors*, 19(10), 2317.
- Jain, A. K., Arora, S. S., Cao, K., Best-Rowden, L., & Bhatnagar, A. (2017). Fingerprint recognition of young children. *IEEE Transactions on Information Forensics and Security*, 12(7), 1501–1514.
- Jain, A. K., Prabhakar, S., & Ross, A. (1999). *Fingerprint matching: Data acquisition and performance evaluation*. Technical Report: MSU TR99-14.
- Jain, A. K., & Ross, A. (2002). Fingerprint mosaicking. In *Proceedings of International Conference on Acoustic Speech and Signal Processing*.
- Jang, J., Elliott, S. J., & Kim, H. (2007). On improving interoperability of fingerprint recognition using resolution compensation based on sensor evaluation. In *Proceedings of International Conference on Biometrics*. LNCS (Vol. 4642, pp. 455–463).
- Jeon, Y. E., Lee, Y. J., Jang, M. K., Seo, B. M., Kang, I. H., Hong, M. T., Lee, J. M., Jacques, E., Mohammed-Brahim, T., & Bae, B. S. (2016). Capacitive sensor array for fingerprint recognition. In *Proceedings of International Conference on Sensing Technology (ICST)* (pp. 1–4).
- Jeon, G., Lee, S., Lee, S. H., Shim, J., Ra, J., Park, K. W., Yeom, H., Nam, Y., Kwon, O., & Park, S. K. (2019). Highly sensitive active-matrix driven self-capacitive fingerprint sensor based on oxide thin film transistor. *Scientific Reports*, 9(1), 3216.
- Jia, X., Yang, X., Zang, Y., Zhang N., & Tian, J. (2012). A cross-device matching fingerprint database from multi-type sensors. In *Proceedings of International Conference on Pattern Recognition (ICPR2012)*, Tsukuba (pp. 3001–3004).
- Jiang, X., Lu, Y., Tang, H., Tsai, J. M., Ng, E. J., Daneman, M. J., Boser, B. E., & Horsley, D. A. (2017). Monolithic ultrasound fingerprint sensor. *Microsystems & Nanoengineering*, 3, 17059.
- Jiang, X., & Ser, W. (2002). Online fingerprint template improvement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8), 1121–1126.
- Jung, S.M., Nam, J. M., Yang, D. H., & Lee, M. K. (2005). A CMOS integrated capacitive fingerprint sensor with 32-bit RISC microcontroller. *IEEE Journal of Solid-State Circuits*, 40(8), 1745–1750.
- Kang, H., Lee, B., Kim, H., Shin, D., & Kim, J. (2003). A study on performance evaluation of fingerprint sensors. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (4th ed., pp. 574–583).
- Kasaei, S., Deriche, M., & Boashash, B. (1997). An efficient quantization technique for wavelet coefficients of fingerprint images. *Signal Processing*, 62(3), 361–366.
- Kasaei, S., Deriche, M., & Boashash, B. (2002). A novel fingerprint image compression technique using wavelets packets and pyramid lattice vector quantization. *IEEE Transactions on Image Processing*, 11(12), 1365–1378.

- Koda, Y., Higuchi, T., & Jain, A. K. (2016). Advances in capturing child fingerprints: A high resolution CMOS image sensor with SLDR method. In *Proceedings of International Conference on Biometrics Special Interest Group (BIOSIG)* (pp. 1–4).
- Krishnasamy, P., Belongie, S., & Kriegman, D. (2011). Wet fingerprint recognition: Challenges and opportunities. In *Proceedings of International Joint Conference on Biometrics (IJCB)*.
- Kumar, A. (2018). *Contactless 3D fingerprint identification*. Springer.
- Kumar, A., & Kwong, C. (2015). Towards contactless, low-cost and accurate 3D fingerprint identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3), 681–696.
- Kwon, D., Yun, I. D., & Lee, S. U. (2010). Rolled fingerprint construction using MRF-based nonrigid image registration. *IEEE Transactions on Image Processing*, 19(12), 3255–3270.
- Lazarick, R., & Wolfhope, P. (2016). Evaluation of ‘non-traditional’ fingerprint sensor performance. In *Proceedings of Symposium on Technologies for Homeland Security (HST)*, Waltham, MA (pp. 1–7).
- Lee, D., Choi, K., Lee, S., & Kim, J. (2003). Fingerprint fusion based on minutiae and ridge for enrollment. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (4th ed., pp. 478–485).
- Lee, D., Choi, K., Choi, H., & Kim, J. (2008). Recognizable-image selection for fingerprint recognition with a mobile-device camera. *IEEE Transaction on Systems, Man, and Cybernetics, Part B*, 38(1), 233–243.
- Lee, H. C., & Gaenslen, R. E. (2012). *Advances in fingerprint technology* (3rd ed.). CRC Press.
- Lee, C., Lee, S., Kim, J., & Kim, S. J. (2006). Preprocessing of a fingerprint image captured with a mobile camera. In *Proceedings of International Conference on Biometrics*. LNCS (Vol. 3832, pp. 348–355).
- Lee, J. W., Min, D. J., Kim, J., & Kim, W. (1999). A 600 dpi capacitive fingerprint sensor chip and image synthesis technique. *IEEE Journal of Solid-State Circuits*, 34(4), 469–475.
- Lepley, M. A. (2001). *JPEG 2000 and WSQ image compression interoperability*. MITRE Technical Report: MTR 00B0000063.
- Liao, Y., Chang, C., Lin, C., You, J., Hsieh, H., Chen, J., Cho, A., Liu, Y., Lai, Y., Tseng, J., Chiang, M., & Lin, Y. (2015). Flat panel fingerprint optical sensor using TFT technology. In *Proceedings of IEEE Sensors Conference*.
- Libert, J., Grantham, J., Bandini, B., Ko, K., Orandi, S., & Watson, C. (2019). *Interoperability assessment 2019: Contactless-to-contact fingerprint capture*. NIST-IR 8307.
- Libert, J. M., Orandi, S., & Grantham, J. D. (2012). *Comparison of the WSQ and JPEG 2000 image compression algorithms on 500 ppi fingerprint imagery*. NIST-IR 7781.
- Liu, F., & Zhang, D. (2014). 3D fingerprint reconstruction system using feature correspondences and prior estimated finger model. *Pattern Recognition*, 47(1), 178–193.
- Liu, F., Zhang, D., Song, C., & Lu, G. (2013). Touchless multiview fingerprint acquisition and mosaicking. *IEEE Transactions on Instrumentation and Measurement*, 62(9), 2492–2502.
- Lorch, H., Morguet, P., & Schroder, H. (2004). Fingerprint distortion measurement. In *Proceedings of Workshop on Biometric Authentication (in ECCV 2004)*. LNCS (Vol. 3087, pp. 111–123).
- Lu, N., Jiang, W., Wu, Q., Geng, D., Li, L., & Liu, M. (2018). A review for compact model of Thin-Film Transistors (TFTs). *Micromachines*, 9(11), 599.
- Lugini, L., Marasco, E., Cukic, B., & Gashi, I. (2013). Interoperability in fingerprint recognition: A large-scale empirical study. In *Proceedings of Conference on Dependable Systems and Networks Workshop*, Budapest, Hungary (pp. 1–6).
- Mainguet, J. G., Pegulu, M., & Harris, J. B. (1999). Fingerchip: Thermal imaging and finger sweeping in a silicon fingerprint sensor. In *Proceedings of Workshop on Automatic Identification Advances Technologies* (pp. 91–94).

- Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L., & Jain, A. K. (2002). FVC2002: Second finger-print verification competition. In *Proceedings of International Conference on Pattern Recognition* (16th ed.).
- Malhotra, A., Chopra, S., Vatsa, M., & Singh, R. (2019). User authentication via finger-selfies. In A. Rattani, R. Derakhshani, & A. Ross (Eds.), *Selfie Biometrics*. Springer.
- Marasco, E., Lugini, L., Cukic, B., & Bourlai, T. (2013). Minimizing the impact of low interoperability between optical fingerprints sensors. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, Arlington, VA (pp. 1–8).
- Marcialis, G. L., & Roli, F. (2004). Fingerprint verification by fusion of optical and capacitive sensors. *Pattern Recognition Letters*, 25(11), 1315–1322.
- Mardiansyah, A. Z., Bejo, A., & Hidayat, R. (2018). Fingerprint image reconstruction for swipe sensor using predictive overlap method. In *Proceedings of MATEC Web of Conferences* (Vol. 154).
- Mathur, S., Vjay, A., Shah, J., Das, S., & Malla, A. (2016). Methodology for partial fingerprint enrollment and authentication on mobile devices. In *Proceedings of International Conference on Biometrics (ICB)*, Halmstad (pp. 1–8).
- Miki, H., & Tsuchitani, S. (2017). Structural design points in arrayed micro thermal sensors (III) ~ polymer-based approach. *International Journal of Engineering and Technical Research*, 7(3), 24–32.
- Modi, S., Elliott, S., & Kim, H. (2009). Statistical analysis of fingerprint sensor interoperability performance. In *Proceedings of International Conference on Biometrics: Theory, Applications, and Systems*, Washington, DC, USA (pp. 1–6).
- Morguet, P., Narr, C., Lorch, H., Wallhoff, F., & Rigoll G. (2004). Reconstruction-free matching for fingerprint sweep sensors. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 1257–1260).
- Morimura, H., Shigematsu, S., & Machida, K. (2000). A novel sensor cell architecture and sensing circuit scheme for capacitive fingerprint sensors. *IEEE Journal of Solid-State Circuits*, 37(10) 1300–1306.
- Nill, N. B. (2006). *Test procedures for verifying image quality requirements for Personal Identity Verification (PIV) single finger capture devices*. MITRE Technical Report, MTR 060170. Retrieved July, 2021, from http://www2.mitre.org/tech/mtf/spec_test.zip.
- Nill, N. B., Lepley, M. A., & Bas, C. F. (2016). *Test procedures for verifying IAFIS image quality requirements for fingerprint scanners and printers, v1.5*. MITRE Technical Report, MTR MTR05B0016R9. Retrieved July, 2021, from http://www2.mitre.org/tech/mtf/spec_test.zip.
- NIST. (2016). *Mobile ID device, best practice recommendation, version 2.1*. NIST Special Publication 500–280 v2.1. Retrieved July, 2021, from <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.500-280v2.1.pdf>.
- NIST. (2020). *Personal identity verification of federal employees and contractors*. Retrieved July, 2021, from <https://csrc.nist.gov/projects/piv>.
- Onyshczak, R., & Youssef, A. (2004). Fingerprint image compression and the wavelet scalar quantization specification. In N. Ratha & R. Bolle (Eds.), *Automatic fingerprint recognition systems*. Springer.
- Orandi, S., Ko, K., Wood, S. S., Grantham, J. D., & Garris, M. D. (2014). *Examination of the impact of fingerprint spatial area loss on matcher performance in various mobile identification scenarios*. NIST-IR 7950.
- Pankanti, S., Prabhakar, S., & Jain, A. K. (2002). On the individuality of fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8), 1010–1025.
- Parziale, G. (2007). Touchless fingerprinting technology. In N. K. Ratha & V. Govindaraju (Eds.), *Advances in Biometrics: Sensors, Algorithms and Systems*. Springer.

- Parziale, G., Diaz-Santana, E., & Hauke, R. (2006). The surround ImagerTM: A multi-camera touchless device to acquire 3D rolled-equivalent fingerprints. In *Proceedings of International Conference on Biometrics*. LNCS (Vol. 3832, pp. 244–250).
- Ramoser, H., Wachmann, B., & Bischof, H. (2002). Efficient alignment of fingerprint images. In *Proceedings of International Conference on Pattern Recognition* (16th ed., Vol. 3, pp. 748–751).
- Ratha, N. K., Connell, J., & Bolle, R. M. (1998). Image mosaicing for rolled fingerprint construction. In *Proceedings of International Conference on Pattern Recognition* (14th ed., Vol. 2, pp. 1651–1653).
- Reed, T., & Meier, R. (1990). Taking dermatoglyphic prints: A self-instruction manual. *American Dermatoglyphics Association Newsletter: Supplement*, 9, 18.
- Ross, A., Dass, S. C., & Jain, A. K. (2006a). Fingerprint warping using ridge curve correspondences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1), 19–30.
- Ross, A., & Jain, A. K. (2004). Biometric sensor interoperability: A case study in fingerprints. In *Proceedings of Workshop on Biometric Authentication (in ECCV 2004)*. LNCS (Vol. 3087, pp. 134–145).
- Ross, A., & Nadgir, R. (2006). A calibration model for fingerprint sensor interoperability. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification III*.
- Ross, A., & Nadgir, R. (2008). A thin-plate spline calibration model for fingerprint sensor interoperability. *IEEE Transaction Data and Knowledge Engineering*, 20(8), 1097–1110.
- Ross, A., Shah, S., & Shah, J. (2006b). Image versus feature mosaicing: A case study in fingerprints. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification III*.
- Rowe, R. K., & Nixon, K. A. (2005). Fingerprint enhancement using a multispectral sensor. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification II*.
- Rowe, R. K., Nixon, K. A., & Butler, P. W. (2007). Multispectral fingerprint image acquisition. In N. K. Ratha & V. Govindaraju (Eds.), *Advances in biometrics: Sensors, algorithms and systems*. Springer.
- Roy, A., Memon, N., & Ross, A. (2017). MasterPrint: Exploring the vulnerability of partial fingerprint-based authentication systems. *IEEE Transactions on Information Forensics and Security*, 12(9), 2013–2025.
- Ryu, C., Han, Y., & Kim, H. (2005). Super-template generation using successive Bayesian estimation for fingerprint enrollment. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (5th ed., pp. 710–719).
- Ryu, C., Kim, H., & Jain, A. K. (2006). Template adaptation based fingerprint verification. In *Proceedings of International Conference on Pattern Recognition* (18th ed., Vol. 4, pp. 582–585).
- Sankaran, A., Malhotra, A., Mittal, A., Vatsa, M., & Singh, R. (2015). On smartphone camera based fingerphoto authentication. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems* (BTAS).
- Saggese, S., Zhao, Y., Kalisky, T., Avery, C., Forster, D., Duarte-Vera, L. E., Almada-Salazar, L. A., Perales-Gonzalez, D., Hubenko, A., Kleeman, M., Chacon-Cruz, E., & Aronoff-Spencer, E. (2019). Biometric recognition of newborns and infants by non-contact fingerprinting: Lessons learned. *Gates Open Research*, 3.
- Sato, N., Machida, K., Morimura, H., Shigematsu, S., Kudou, K., Yano, M., & Kyuragi, H. (2003). MEMS fingerprint sensor immune to various finger surface conditions. *IEEE Transactions on Electron Devices*, 50(4), 1109–1116.
- Sato, N., Shigematsu, S., Morimura, H., Yano, M., Kudou, K., Kamei, T., & Machida, K. (2005). Novel surface structure and its fabrication process for MEMS fingerprint sensor. *IEEE Transactions on Electron Devices*, 52(5), 1026–1032.
- Schneider, J. K. (2007). Ultrasonic fingerprint sensors. In N. K. Ratha & V. Govindaraju (Eds.), *Advances in biometrics: Sensors, algorithms and systems*. Springer.

- Schneider, J. K., Richardson, C. E., Kiefer, F. W., & Govindaraju, V. (2003). On the correlation of image size to system accuracy in automatic fingerprint identification systems. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (4th ed., pp. 895–902).
- Schneider, J., & Wobschall, D. (1991). Live scan fingerprint imagery using high resolution C-SCAN ultrasonography. In *Proceedings of International Carnahan Conference on Security Technology* (25th ed., pp. 88–95).
- Seo, W., Pi, J., Cho, S. H., Kang, S., Ahn, S., Hwang, C., Jeon, H., Kim, J. & Lee, M. (2018). Transparent fingerprint sensor system for large flat panel display. *Sensors*, 18(1).
- Setlak, D. R. (1999). *Electric field fingerprint sensor apparatus and related methods*. US Patent 5963679.
- Setlak, D. S. (2004). Advances in fingerprint sensors using RF imaging techniques. In N. Ratha & R. Bolle (Eds.), *Automatic fingerprint recognition systems* (pp. 27–53). Springer.
- Setlak, D. R., VanVonno, N. W., Newton, M., & Salatino, M. M. (2000). *Fingerprint sensor including an anisotropic dielectric coating and associated methods*. US Patent 6088471.
- Sha, L., Zhao, F., & Tang, X. (2007). A two-stage fusion scheme using multiple fingerprint impressions. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 385–388).
- Skodras, A., Christopoulos, C., & Ebrahimi, T. (2001). JPEG 2000 still image compression standard. *IEEE Signal Processing Magazine*, 18(5), 36–58.
- Sousedik, C., & Breithaupt, R. (2017). Full-fingerprint volumetric subsurface imaging using Fourier-domain optical coherence tomography. In *2017 5th International Workshop on Biometrics and Forensics (IWBF)* (pp. 1–6).
- Stein, C., Nickel, C., & Busch, C. (2012). Fingerphoto recognition with smartphone cameras. In *Proceedings of International Conference of Biometrics Special Interest Group (BIOSIG)* (pp. 1–12).
- Tabei, J., Sasajima, H., & Mori, T. (2016). Epoxy molding compound for fingerprint sensor. In *Proceedings of International Conference on Electronics Packaging (ICEP)* (pp. 553–556).
- Tang, H., Lu, Y., Fung, S., Tsai, J. M., Daneman, M., Horsley, D. A., & Boser, B. E. (2015). Pulse-echo ultrasonic fingerprint sensor on a chip. In *Proceedings of International Conference on Solid-State Sensors, Actuators and Microsystems*, Anchorage, AK (pp. 674–677).
- Tang, H., Lu, Y., Jiang, X., Ng, E. J., Tsai, J. M., Horsley, D. A., & Boser, B. E. (2016). 3-D ultrasonic fingerprint sensor-on-a-chip. *IEEE Journal of Solid-State Circuits*, 51(11), 2522–2533.
- Tartagni, M., & Guerrieri, R. (1998). A fingerprint sensor based on the feedback capacitive sensing scheme. *IEEE Journal of Solid-State Circuits*, 33(1), 133–142.
- Thomas, D. A., & Bryant, F. R. (2000). *Electrostatic discharge protection for integrated circuit sensor passivation*. US Patent 6091082.
- Toh, K. A., Yau, W. Y., Jiang, X., Chen, T. P., Lu, J., & Lim, E. (2001). Minutiae data synthesis for fingerprint identification applications. In *Proceedings of International Conference on Image Processing*.
- Tordera, D., Peeters, B., Akkerman, H. B., van Breemen, A. J. J. M., Maas, J., Shanmugam, S., Kroonenijer, A. J., & Gelinck, G. H. (2019). A high resolution thin film fingerprint sensor using a printed organic photodetector. *Advanced Material Technologies*, 4(11).
- Tsikos, C. (1982). *Capacitive fingerprint sensor*. US Patent 4353056.
- Uz, T., Bebis, G., Erol, A., & Prabhakar, S. (2009). Minutiae-based template synthesis and matching for fingerprint authentication. *Computer Vision and Image Understanding*, 113(9), 979–992.
- Viterbi, A. J. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13, 260–269.
- Wang, L., El-Maksoud, R. H. A., Sasian, J. M., & Valencia, V. S. (2009). Illumination scheme for high-contrast, contactless fingerprint images. *Proceedings of SPIE Novel Optical Systems Design and Optimization*, 7429(1).

- Wang, Y., Hassebrook, L. G., & Lau, D. L. (2010). Data acquisition and processing of 3-D fingerprints. *IEEE Transactions on Information Forensics and Security*, 5(4), 750–760.
- Wang, Y., Kong, X., Wang, R., Jin, C., & Kim, H. (2018). Study and realization of partial fingerprint mosaicking technology for mobile devices. In *Proceedings of Chinese Conference on Biometric Recognition*, Cham.
- Watson, C. I. (1993). *NIST Special Database 14, Fingerprint Database*. U.S. National Institute of Standards and Technology.
- Wei, P., Marathe, S., Zhou, J., & Pommerenke, D. (2017). ESD susceptibility evaluation on capacitive fingerprint module. In *Proceedings of International Symposium on Electromagnetic Compatibility & Signal/Power Integrity (EMCSI)*, Washington, DC (pp. 175–180).
- Weingaertner, D., Bellon, O., Silva, L., & Cat, M. (2008). Newborn's biometric identification: Can it be done? In *Proceedings of International Conference on Computer Vision Theory and Applications* (pp. 200–205).
- Wilson, C. L., Watson, C. I., & Paek, E. G. (2000). Effect of resolution and image quality on combined optical and neural network fingerprint matching. *Pattern Recognition*, 33(2), 317–331.
- Xia, X., & O'Gorman, L. (2003). Innovations in fingerprint capture devices. *Pattern Recognition*, 36(2), 361–369.
- Yang, C., & Zhou, J. (2006). A comparative study of combining multiple enrolled samples for fingerprint verification. *Pattern Recognition*, 39(11), 2115–2130.
- Yau, W. Y., Chen, T. P., & Morguet, P. (2004). Benchmarking of fingerprint sensors. In *Proceedings of Workshop on Biometric Authentication (in ECCV 2004)*. LNCS (Vol. 3087, pp. 89–99).
- Yau, W. Y., Toh, K. A., Jiang, X., Chen, T. P., & Lu, J. (2000). On fingerprint template synthesis. In *Proceedings of International Conference on Control Automation Robotics and Vision* (6th ed.).
- Yeung, H. W., Moon, Y. S., & Chan, K. C. (2004). Fingerprint registration for small fingerprint sensors. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification I*.
- Yin, X., Zhu, Y., & Hu, J. (2021). 3D fingerprint recognition based on ridge-valley-guided 3D reconstruction and 3D topology polymer feature extraction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(3), 1085–1091.
- Young, N. D., Harkin, G., Bunn, R. M., McCulloch, D. J., Wilks, R. W., & Knapp, A. G. (1997). Novel fingerprint scanning arrays using polysilicon tft's on glass and polymer substrates. *IEEE Electron Device Letters*, 18(1), 19–20.
- Zang, Y., Yang, X., Jia, X., Zhang, N., Tian, J., & Zhao, J. (2013). Evaluation of minutia cylinder-code on fingerprint cross-matching and its improvement with scale. In *Proceedings of International Conference on Biometrics (ICB)*, Madrid, Spain (pp. 1–8).
- Zhang, Y. L., Yang, J., & Wu, H. T. (2005). A hybrid swipe fingerprint mosaicing scheme. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (5th ed., pp. 131–140).
- Zhang, Y. L., Yang, J., & Wu, H. T. (2006a). Sweep fingerprint sequence reconstruction for portable devices. *Electronics Letters*, 42(4), 204–205.
- Zhang, Y. L., Yang, J., & Wu, H. T. (2006b). Coarse-to-fine image registration for sweep fingerprint sensors. *Optical Engineering*, 45(6).
- Zhang, Y., Xiao, G., Li, Y., Wu, H., & Huang, Y. (2010). Slap fingerprint segmentation for live-scan devices and ten-print cards. In *Proceedings of 20th International Conference on Pattern Recognition*, Istanbul (pp. 1180–1183).
- Zhang, Y., Fang, S., Bian, Y., & Li, Y. (2013). Real-time rolled fingerprint construction based on key-column extraction. In *Proceedings of Chinese Conference on Biometric Recognition*.

- Zhao, S., & Wang, X. (2009). Fingerprint Image Compression Based on Directional Filter Banks and TCQ. In *Proceedings of International Workshop on Knowledge Discovery and Data Mining*, Moscow (pp. 660–663).
- Zhou, G., Qiao, Y., & Mok, F. (1998). *Fingerprint sensing system using a sheet prism*. US Patent 5796858.
- Zhou, J., He, D., Rong, G., & Bian, Z. (2001). Effective algorithm for rolled fingerprint construction. *Electronics Letters*, 37(8), 492–494.



Fingerprint Analysis and Representation

3

Abstract

This chapter introduces classical domain knowledge-based and emerging learning-based techniques for feature extraction in fingerprints. Specific sections are dedicated to explain the most effective approaches for segmentation, local orientation and frequency extraction, singularity detection and pose estimation, image enhancement, and minutiae and pore detection. The computation of global and local fingerprint image quality is also reviewed. Particular emphasis is given to robust feature extraction algorithms, trained on large corpuses of real or synthetic data, that can reliably operate on images of various quality encountered in applications.

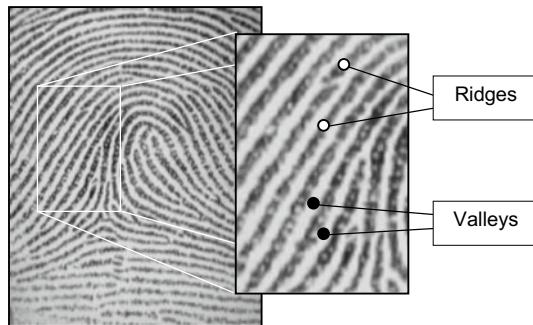
Keywords

Feature extraction • Pose estimation • Local ridge orientations • Local ridge frequencies • Segmentation • Ridge flow singularities • Enhancement • Minutiae • Pores • Fingerprint quality

3.1 Introduction

A fingerprint is the impression of the exterior appearance of the fingertip epidermis. The most evident structural characteristic of a fingerprint is a pattern of interleaved *ridges* and *valleys* (Ashbaugh, 1999; Hicklin, 2009); in a fingerprint image, ridges (also called ridge lines) are dark whereas valleys are bright (see Fig. 3.1). Ridges vary in width from 100 µm, for very thin ridges, to 300 µm for thick ridges. Generally, the period of a ridge/valley cycle is about 500 µm. Most injuries to a finger such as superficial burns, abrasions, or cuts do not affect the underlying ridge structure, and the original pattern is duplicated in any new skin that grows.

Fig. 3.1 Ridges and valleys in a fingerprint image

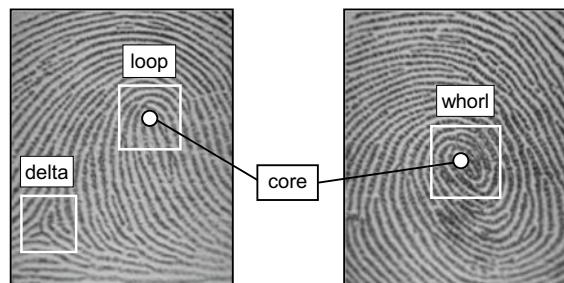


Ridge details are generally described in a hierarchical order at three different levels, namely Level 1 (the overall global ridge flow pattern), Level 2 (minutiae points), and Level 3 (pores, the local shape of ridge edges, etc.).

At the global level (Level 1), ridges often run smoothly in parallel but exhibit one or more regions where they assume distinctive shapes (characterized by high curvature, frequent ridge terminations, etc.). These regions, called *singularities* or *singular regions*, may be broadly classified into three typologies: *loop*, *delta*, and *whorl* (see Fig. 3.2). Singular regions belonging to loop, delta, and whorl types are typically characterized by \cap , Δ , and O shapes, respectively. Sometimes whorl singularities are not explicitly introduced because a whorl type can be described in terms of two loop singularities facing each other.

Fingerprint matching algorithms can pre-align fingerprint images according to a landmark or a center point, called the *core*. Henry (1900) defined the core point as “the northmost point of the innermost ridge line.” In practice, the core point corresponds to the center of the northmost loop-type singularity. For fingerprints that do not contain loop or whorl singularities (e.g., those belonging to the Arch class in Fig. 3.3), it is difficult to define the core. In these cases, the core is usually associated with the point of maximum ridge-line curvature. Unfortunately, due to the high variability of fingerprint patterns, it is difficult to reliably locate a registration (core) point in all the fingerprint images. Singular regions are commonly used for fingerprint classification (see Fig. 3.3), that is assigning a

Fig. 3.2 Singular regions (white boxes) and core points (filled circles) in fingerprint images



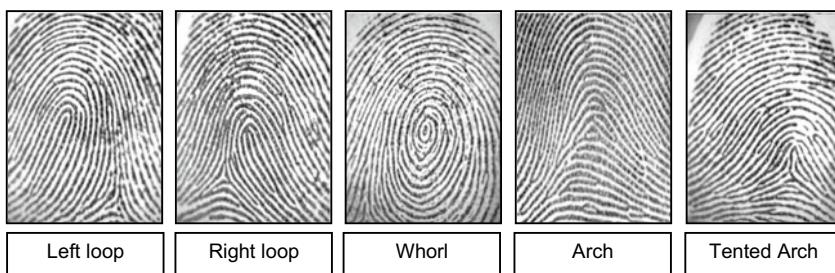


Fig. 3.3 One fingerprint from each of the five major classes defined by Henry (1900)

fingerprint to a class among a set of distinct classes, with the aim of simplifying search and retrieval (ref. Chap. 5).

At the local level (Level 2), other important features, called *minutiae*, can be found in the fingerprint patterns. Minutiae means small detail; in the context of fingerprints, it refers to various ways that the ridges can be discontinuous (see Fig. 3.4). For example, a ridge can suddenly come to an end (ridge ending) or can divide into two ridges (bifurcation). Minutiae are the most commonly used features in both manual and automated fingerprint matching. Sir Francis Galton (1822–1911) was the first person to categorize minutiae and to observe that they remain unchanged over an individual's lifetime (Galton, 1892). Minutiae are sometimes called "Galton details" in his honor. In a full fingerprint (i.e., rolled impression), the number of minutiae can be more than 100; however, as discussed in Chap. 8, the spatial and angular coincidence or correspondence of a relatively small number of minutiae (e.g., as few as 12–15) may be sufficient to claim with confidence that two fingerprint impressions originate from the same finger. Some interesting statistical data on minutiae distribution can be found in Champod et al. (2016) and Stoney and Thornton (1987); in particular, average densities of 0.49 and 0.18 minutiae/mm² were estimated by Champod et al. (2016) inside the singular regions and outside the singular regions, respectively. Although several types of minutiae can be defined (the most common types

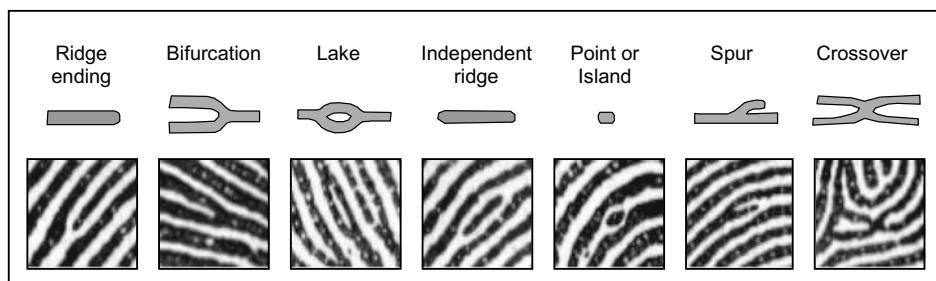


Fig. 3.4 Seven most common minutiae types

are shown in Fig. 3.4), only *ridge endings* and *bifurcations* are considered by the fingerprint encoding standards (ISO/IEC 19794-2, 2011 and ANSI/NIST-ITL 1-2011, 2015) to deal with the practical difficulty in automatically discerning the different types with high accuracy. ANSI/NIST-ITL 1-2011 (2015) recommends that all complex minutiae types such as crossovers/trifurcations should be marked as combinations of bifurcations and ridge endings.

Each minutia is denoted by its origin (i.e., x - and y -coordinates), angle, type, and quality. The angle is defined according to the ridge/valley orientation at the minutia position (Fig. 3.5). In practice, an ambiguity exists between ridge-ending and bifurcation minutiae types; depending on the finger pressure against the surface where the fingerprint impression is formed, ridge endings may appear as bifurcations and vice versa. However, given the convention used to define minutiae angle, there is no significant change in the angle if the minutia appears as a ridge ending in one impression and as a bifurcation in another impression of the same finger.

Figure 3.6a shows a portion of the fingerprint image where the ridge lines appear as dark traces on a light background; two ridge endings (1, 2) and one bifurcation (3) are shown. Note that on the negative image (Fig. 3.6b), the corresponding minutiae take the same positions, but their type is exchanged: ridge endings now appear as bifurcations

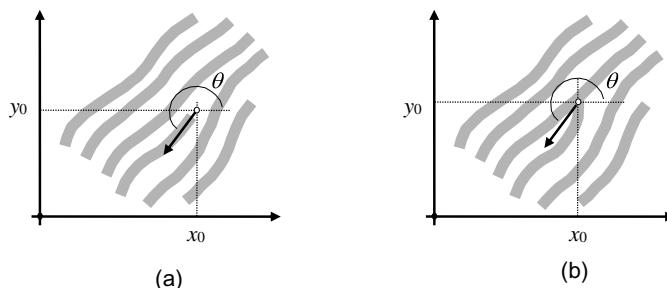
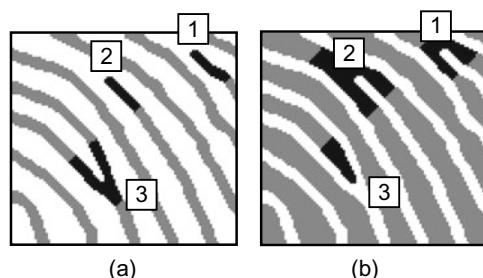


Fig. 3.5 Minutiae encoding in ISO/IEC 19794-2 (2011) and ANSI/NIST-ITL 1-2011 (2015). **a** For a ridge-ending minutia, the origin $[x_0, y_0]$ is placed at the forking point of the medial skeleton of the valley area immediately in front of the ridge ending, and θ is the angle (running up the ridge) that the ridge tangent forms with the horizontal axis; **b** for a bifurcation minutia, the origin is placed at the forking point of the medial skeleton of the ridge and θ is directed to the valley. Note that detecting a ridge ending as a bifurcation (or vice versa) does not cause an abrupt 180° change in the angle

Fig. 3.6 The ridge-ending/bifurcation duality; in **a** a binary-ridge map image and **b** its negative image (i.e., dark and bright pixels are swapped)



and vice versa (this property is known as ridge-ending/bifurcation *duality*). Besides the two coordinates and the angle, other attributes can be associated with each minutia: these usually consist of features extracted from the minutia neighbors and, as discussed in Sect. 4.4, can be very useful to improve fingerprint matching accuracy.

At the very local level (Level 3), additional fine details can be extracted in the fingerprint pattern. They include all dimensional attributes of the ridges such as width, shape, edge contour (Fig. 3.7a), pores (Fig. 3.7a), incipient ridges (Fig. 3.7b), breaks, creases, and scars (Fig. 3.7c). Each ridge of the epidermis (outer skin) is dotted with *pores* (or *sweat pores*) along its entire length and anchored to the dermis (inner skin) by a double row of peglike protuberances or papillae (Roddy & Stosz, 1997). Pores may range in size from 60 to 250 μm . It was observed that the number of pores along a centimeter of ridge varies from 9 to 18. It has been claimed that 20–40 pores may be sufficient to determine the identity of a person (Ashbaugh, 1999). Although Level-3 features are highly distinctive and extremely important for latent fingerprint examiners, currently few automated matching techniques use them since their reliable detection even in high-resolution fingerprint scanners (e.g., 1,000 dpi) and good-quality fingerprint images is not guaranteed (Zhang et al., 2011a; Zhao & Jain, 2010). Thanks to the work of the Committee to Define an Extended Fingerprint Feature Set (CDEFFS, 2008) Level-3 features are now encoded in Type-9 records of ANSI/NIST-ITL 1-2011 (2015).

Although a few fingerprint matching techniques in the literature directly compare images through correlation-based methods (see Sect. 4.2), a representation based on the sensed gray-scale image intensities is usually considered not robust. Therefore, most of the fingerprint recognition algorithms employ a feature extraction stage for identifying salient features.

The features extracted from fingerprint images often have a direct physical counterpart (e.g., singularities or minutiae), but sometimes they are not directly related to any physical

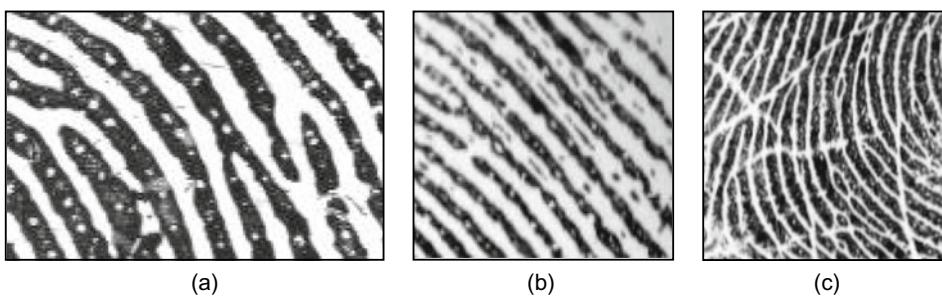


Fig. 3.7 Level-3 fingerprint features. **a** A fingerprint portion, acquired at 1000 dpi, where pores are well evident. The local variability of the ridge width and shape and the irregularity of the ridge contours is also visible; **b** incipient ridges are partially developed ridges that can occur in the valley between normal ridges: they are often fragmented and do not contain pores; and **c** creases present in a portion of a fingerprint

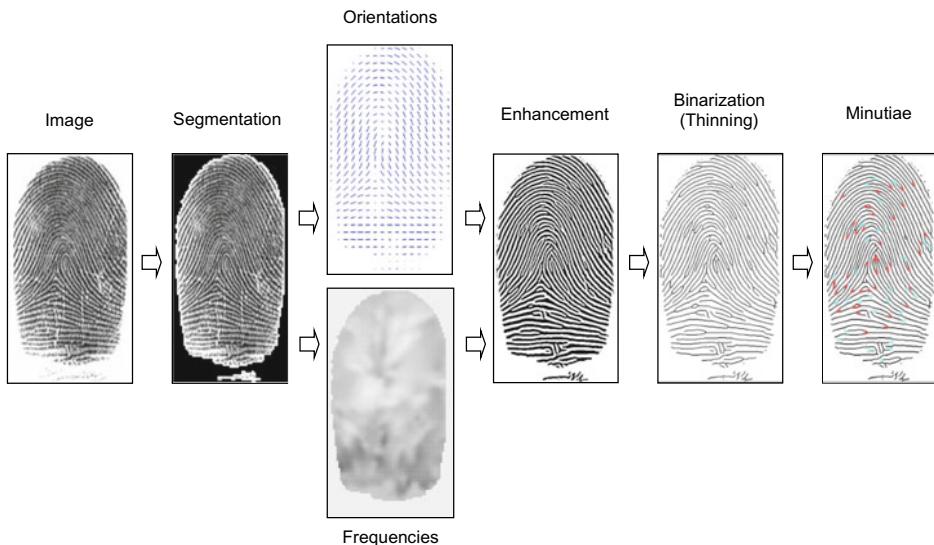


Fig. 3.8 Processing steps in a typical minutiae extraction pipeline. Details of individual steps are introduced in the following sections

attributes. This chapter mainly discusses the extraction of three-level features with clear physical meanings; other fingerprint representations used by automated matching techniques (e.g., minutiae descriptors, SIFT, and CNN-based representations) are introduced in Chap. 4. Features may be used either for matching or their computation may serve as an intermediate step for the derivation of other features. Figure 3.8 shows the typical processing steps performed to extract minutiae.

Throughout this book, a fingerprint image is represented as a two-dimensional surface. Let \mathbf{I} be a gray-scale fingerprint image with g gray levels, and $\mathbf{I}[x, y]$ be the gray level of pixel $[x, y]$ in \mathbf{I} . Let $z = S(x, y) = (g - 1 - \mathbf{I}[x, y])$ be the discrete surface corresponding to the image \mathbf{I} . By associating dark pixels with gray levels close to 0 and bright pixels with gray levels close to $g - 1$, the fingerprint ridge lines (appearing dark in \mathbf{I}) correspond to surface ridges, and the spaces between the ridge lines (appearing bright in \mathbf{I}) correspond to surface valleys (Fig. 3.9).

3.2 Segmentation

The term fingerprint *segmentation* is generally used to denote the separation of ridge–valley area (foreground) from the image background; an example of segmentation is shown in Fig. 3.10. Separating the background is useful to avoid the extraction of features in noisy areas that is often the background. Some authors use the term segmentation to indicate

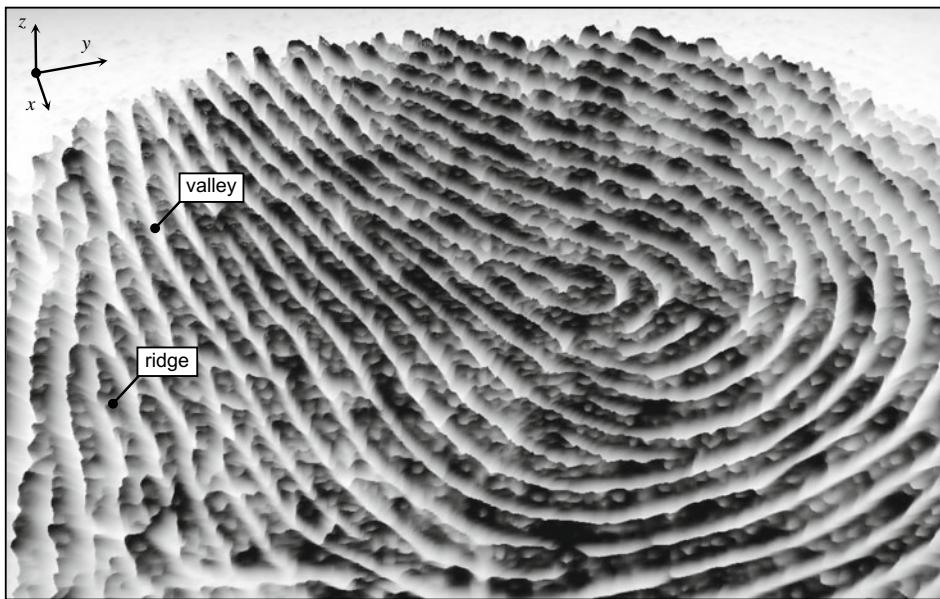


Fig. 3.9 A surface S representing a fingerprint portion

the transformation of the fingerprint image from gray-scale to binary image; throughout this book, the latter processing is referred to as *fingerprint binarization*.

Because fingerprint images are striated patterns (dark and light bands), using a global or local thresholding technique (Gonzales & Woods, 2007) does not allow the fingerprint area to be effectively isolated from the background. In fact, what really discriminates foreground and background is not the average image intensity but the presence of a striped and oriented pattern in the foreground and of an isotropic pattern (i.e., which does not have a dominant orientation) in the background. If the image background were always uniform and lighter than the fingerprint area, a simple approach based on local intensity thresholding could be effective for discriminating foreground and background; in practice, the presence of noise (such as that produced by dust and grease on the finger surface or surface of live-scan fingerprint scanners) requires more robust segmentation techniques.

Another problem related to fingerprint segmentation is *separating overlapped fingerprints*. This is of particular interest in forensic analysis to isolate two or more overlapped latent fingerprints unintentionally left on the same object. More information on this problem is provided in Chap. 6; the interested reader is also referred to Chen et al. (2011a), Feng et al. (2012), Zhao and Jain (2012), Zhang et al. (2014), Branka et al. (2019).

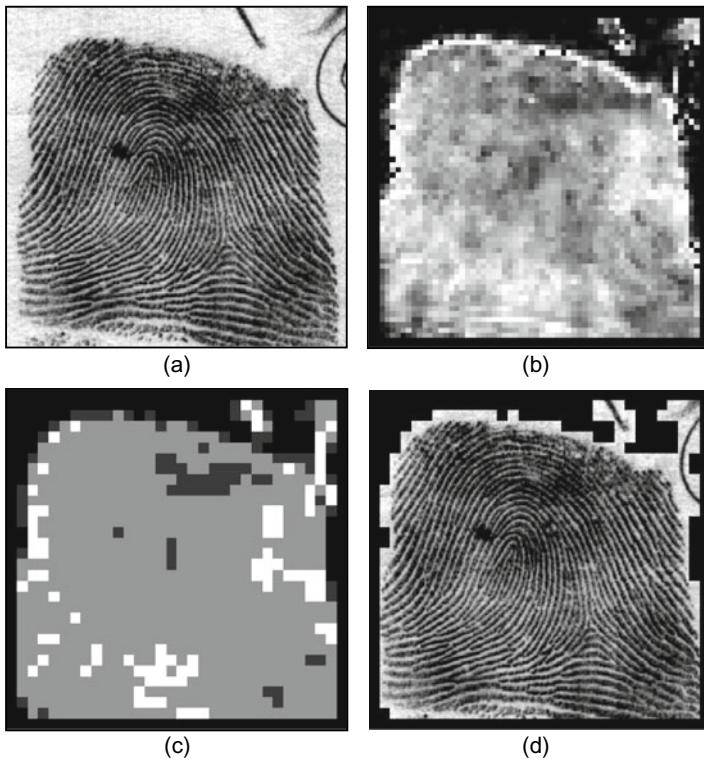


Fig. 3.10 Segmentation of a fingerprint image as proposed by Rath et al. (1995): **a** original image; **b** variance field; **c** quality image derived from the variance field: a quality value “good,” “medium,” “poor”, or “background” is assigned to each block according to its variance; and **d** segmented image. © Elsevier. Reprinted, with permission, from Rath et al. (1995)

3.2.1 Segmentation Based on Handcrafted Features and Thresholding

To discriminate between striated patterns in the foreground and isotropic patterns in the background, a number of (block-wise) features have been proposed as follows:

- The presence of peaks in local histograms of ridge orientations (Mehtre et al., 1987).
- The variance of gray levels in the orthogonal direction to the ridge orientation (Ratha et al., 1995). See Fig. 3.10.
- The average magnitude of the gradient in each image block (Maio & Maltoni, 1997).
- The variance of the Gabor filter responses (Shen et al., 2001; Alonso-Fernandez et al., 2005; and Wang et al., 2005).
- The local energy in the Fourier spectrum (Pais Barreto Marques & Thome, 2005; Chikkerur et al., 2007).

Once the features have been extracted, a simple global thresholding is usually effective for the segmentation. Most of the above methods are very fast and perform reasonably well when the background is uniform and not too noisy. The methods introduced in the following subsections allow to achieve better results on difficult cases but can be practically adopted when the associated computational resource is feasible for the intended application.

3.2.2 Learning-Based Segmentation with Simple Classifiers

Learning-based techniques were introduced to obtain more accurate segmentations with respect to approaches based on feature value thresholding:

- Bazen and Gerez (2001) proposed a pixel-wise method, where three features (gradient coherence, intensity mean, and intensity variance) are computed in the neighborhood of each pixel, and a linear classifier associates the pixel with the background or the foreground. A supervised technique is used to learn the optimal parameters for the linear classifier for each specific fingerprint sensor. A final morphological post-processing step (Gonzales & Woods, 2007) is performed to eliminate holes in both the foreground and background and to regularize the external silhouette of the fingerprint area. The same feature vectors were used by Yin et al. (2005) for point-wise segmentation, but adopting a quadratic separation surface (non-linear classifier) instead of a hyperplane (linear classifier) causes a relevant reduction of the pixel classification rate.
- Chen et al. (2004a) trained a linear classifier to select foreground blocks based on (i) the block clusters degree, (ii) the difference of local block intensity mean and global image intensity mean, and (iii) the block variance. The block cluster degree is a measure of clustering of the ridge (dark) and valley (bright) gray levels. Morphology is then applied during post-processing to regularize the results. Some examples of segmentation are shown in Fig. 3.11.
- The basics of Zhu et al. (2006) technique are discussed in Sect. 3.3.4 in the context of local orientation correction.
- Paiva and Tasdizen (2012) argued that fingerprint patches seen in a high-dimensional space form a simple and highly regular circular manifold. Therefore, they introduced some features to characterize the manifold topology and trained a Fisher linear classifier (Duda et al., 2000) to discriminate between foreground and background blocks.



Fig. 3.11 Some examples of segmentation (marked by the solid boundary) with the method proposed by Chen et al. (2004a). © Springer Nature. Reprinted, with permission, from Chen et al. (2004a)

3.2.3 Total Variation Models

Total Variation (TV) models have been widely used in the context of image decomposition (Aujol et al., 2006; Buades et al., 2010). Some researchers successfully applied these techniques to latent fingerprints (Zhang et al., 2013; Cao et al., 2014), whose segmentation is particularly challenging and usually assisted by forensic experts through manual markup of the ROI (Region of Interest) as discussed in Chap. 6.

The TV models decompose the input image into two layers: cartoon and texture. In the context of fingerprint segmentation, the cartoon layer typically contains structured noise (drawings, characters, stain, speckle, etc., in the background region), while the texture layer contains the oscillatory or textured component characterizing the fingerprint pattern (foreground). The cartoon–texture decomposition facilitates the process of segmentation, as the region of interest can be easily detected from the texture layer using traditional segmentation methods.

The methods proposed by Zhang et al. (2012a, b, 2013) extend the classical TV methods by locally adjusting the cartoon/texture relative strength according to the background noise level and by imposing directional information. A three-layer decomposition (cartoon, texture, and noise) was proposed by Thai and Gottschlich (2016) whose method enforces sparsity and smoothness in the texture layer. As shown in Fig. 3.12, TV methods are quite effective to deal with difficult segmentation cases; on the other hand, the iterative approaches used to solve the underlying optimization problem are computationally demanding.

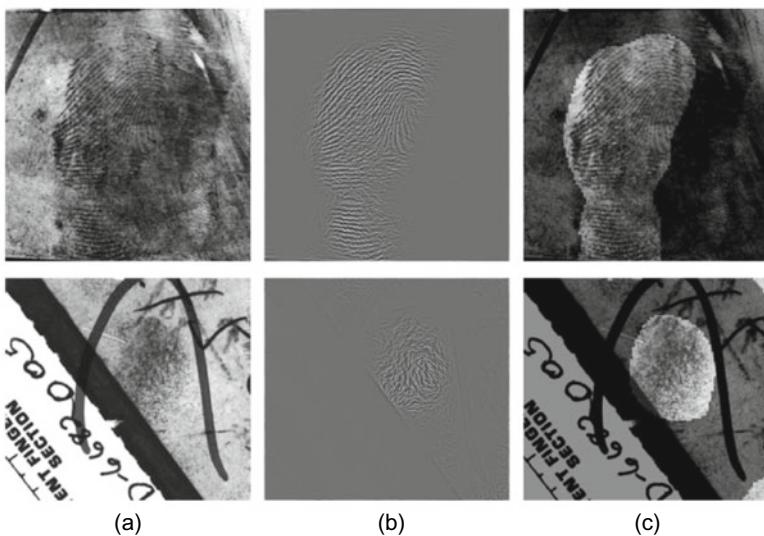


Fig. 3.12 Some examples of latent fingerprint image segmentation with the method proposed by Zhang et al. (2013). **a** Input image, **b** the texture layer after the decomposition, and **c** the segmentation mask imposed over the image. © IEEE. Reprinted, with permission, from Zhang et al. (2013)

3.2.4 Deep Learning Models

Deep learning approaches have been proposed for latent fingerprint segmentation by Zhu et al. (2017), Ezeobiejesi and Bhanu (2017), and Nguyen et al. (2018a):

- Zhu et al. (2017) trained four CNNs by using multi-scale patches and fused the corresponding output scores to improve segmentation accuracy.
- Ezeobiejesi and Bhanu (2017) used a stack of restricted Boltzmann machines (RBMs) to implement a generative feature learning model. For each fingerprint patch, the extracted features are then passed to a simple binary classifier.

Both these methods process patches separately (i.e., through a sliding window technique), and this makes them computationally inefficient. On the other hand, the Nguyen et al. (2018a) method combines fully convolutional neural network and detection-based approaches to process the entire input latent image in one shot. A visual attention mechanism was specifically designed to focus only on the latent fingerprint regions. More details on this approach are reported in Sect. 6.4.3.

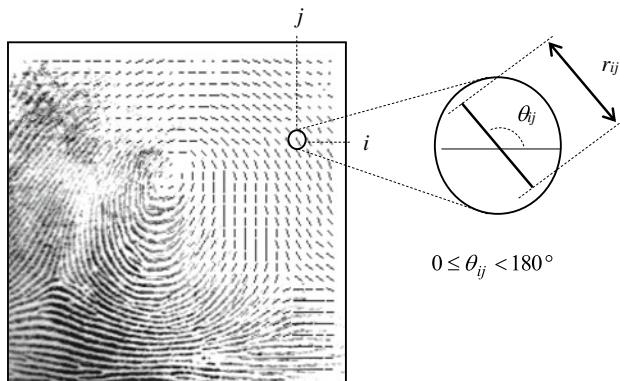
3.3 Local Ridge Orientation Estimation

The estimation of local ridge orientations, also called ridge flow, is one of the most important steps of fingerprint image processing. In fact, the availability of reliable orientations can greatly improve subsequent tasks (in particular, fingerprint enhancement by contextual filtering).

The local ridge orientation at a pixel $[x, y]$ is the angle θ_{xy} that the fingerprint ridges, crossing through an arbitrary small neighborhood centered at $[x, y]$, form with the horizontal axis. Because fingerprint ridges are not directed, θ_{xy} is an unoriented direction lying in $[0\dots180^\circ[$. In the rest of the book, we use the term *orientation* to denote an unoriented direction in $[0\dots180^\circ[$ and the term *direction* to indicate an oriented direction in $[0\dots360^\circ[$.

Instead of computing local ridge orientation at each pixel, most of the fingerprint processing and feature extraction methods estimate the local ridge orientation at discrete positions (this reduces computational efforts and still allows estimates at other pixels through interpolation). The fingerprint *orientation field* (also called *orientation image* or *directional image*), first introduced by Grasselli (1969), is a matrix \mathbf{D} whose elements encode the local orientation of the fingerprint ridges. Each element θ_{ij} , corresponding to the node $[i, j]$ of a grid located over the pixel $[x_i, y_j]$, denotes the average orientation of the fingerprint ridges in a neighborhood of $[x_i, y_j]$ (see Fig. 3.13). An additional value r_{ij} is often associated with each element θ_{ij} to denote the reliability (or consistency) of the orientation. The value r_{ij} is low for noisy and seriously corrupted regions and high for good-quality regions in the fingerprint image.

Fig. 3.13 A fingerprint image faded into the corresponding orientation image computed over a square-meshed grid of size 16×16 . Each element denotes the local orientation of the fingerprint ridges; the element length is proportional to its reliability



3.3.1 Gradient-Based Approaches

The simplest and most natural approach for extracting local ridge orientation is based on the computation of gradients in the fingerprint image. The gradient $\nabla(x, y)$ at point $[x, y]$ of \mathbf{I} is a two-dimensional vector $[\nabla_x(x, y), \nabla_y(x, y)]$, where ∇_x and ∇_y components are the derivatives of \mathbf{I} at $[x, y]$ with respect to the x - and y -directions, respectively. It is well known that the gradient phase angle denotes the direction of the maximum intensity change. Therefore, the direction θ of a hypothetical edge that crosses the region centered at $[x, y]$ is orthogonal to the gradient phase angle at $[x, y]$. This method, although simple and efficient, has some drawbacks. First, using the classical Prewitt or Sobel convolution masks (Gonzales & Woods, 2007) to determine ∇_x and ∇_y components of the gradient and computing θ according to the arctangent of the ∇_y/∇_x ratio present problems due to the non-linearity and discontinuity around 90° . Second, a single orientation estimate reflects the ridge–valley orientation at too fine a scale and is generally very sensitive to the noise in the fingerprint image; on the other hand, simply averaging gradient estimates is not meaningful due to the circularity of angles: the average orientation between 5° and 175° is not 90° (as an arithmetic average would suggest) but 0° . Furthermore, the concept of average orientation is not always well defined; consider the two orthogonal orientations 0° and 90° ; is the correct average orientation 45° or 135° ?

Kass and Witkin (1987) proposed a simple but elegant solution to the above problem, which allows local gradient estimates to be averaged. Their basic idea is to double the angles, so that each single orientation estimate is encoded by the vector:

$$\mathbf{d} = [r \cdot \cos(2\theta), r \cdot \sin(2\theta)], \quad (3.1)$$

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

$$\bar{\mathbf{d}} = \left[\frac{1}{n^2} \sum_W r \cdot \cos(2\theta), \frac{1}{n^2} \sum_W r \cdot \sin(2\theta) \right]. \quad (3.2)$$

Computing the average between two orthogonal orientations with Eq. (3.2) involves summing two vectors facing each other, and therefore, the length of the resulting vector is zero. This indicates that the vector is meaningless, independent of its orientation.

Based on the above idea, an effective method may be derived for computing the fingerprint orientation image (Bazen & Gerez, 2002; Rao, 1990; Ratha et al., 1995). For example, Ratha et al. (1995) computed the dominant ridge orientation θ_{ij} by combining multiple gradient estimates within a 17×17 window W centered at $[x_i, y_j]$:

$$\theta_{ij} = 90^\circ + \frac{1}{2} \text{atan2}(2G_{xy}, G_{xx} - G_{yy}), \quad (3.3)$$

$$G_{xy} = \sum_{h=-8}^8 \sum_{k=-8}^8 \nabla_x(x_i + h, y_j + k) \cdot \nabla_y(x_i + h, y_j + k),$$

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

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

where ∇_x and ∇_y are the x - and y -gradient components computed through 3×3 Sobel masks, and $\text{atan2}(y,x)$ calculates the arctangent of the two variables y and x : it is similar to calculating the arctangent of y/x , except that the signs of both arguments are used to determine the quadrant of the result. An example of local orientation image computed with Eq. (3.3) is shown in Fig. 3.14b. Bazen and Gerez (2002) have shown that this method is mathematically equivalent to the principal component analysis of the autocorrelation matrix of the gradient vectors. Another gradient-based method, independently proposed by Donahue and Rokhlin (1993), relies on least-squares minimization to perform the averaging of orientation estimates and leads to equivalent expressions.

The reliability r of the estimate θ can be derived by the concordance (or coherence) of the orientation vectors \mathbf{d} in the local window W (Bazen & Gerez, 2002; Kass & Witkin, 1987). In fact, due to the continuity and smoothness of fingerprint ridges, sharp orientation

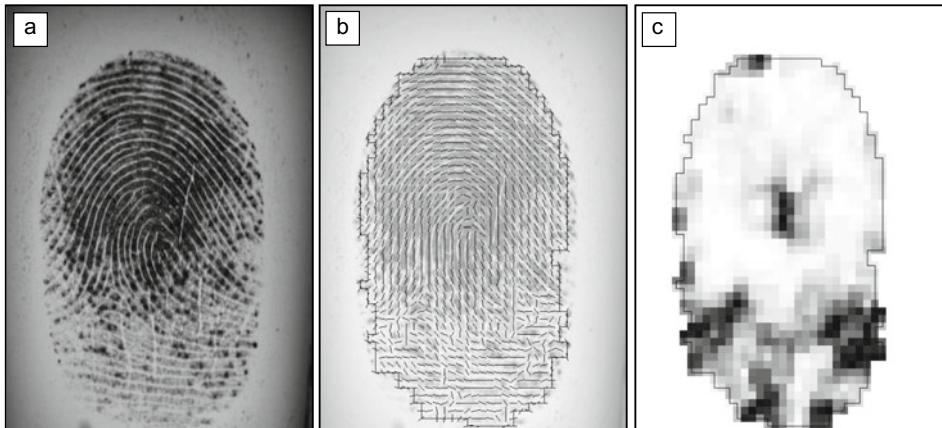


Fig. 3.14 A fingerprint image **a** local ridge orientation field, **b** computed with Eq. (3.3) and its local coherence map, and **c** computed with Eq. (3.4) over 3×3 blocks

changes often denote unreliable estimation. Kass and Witkin (1987) define the coherence as the norm of the sum of orientation vectors divided by the sum of their individual norms; this scalar always lies in [0, 1]: its value is 1 when all the orientations are parallel to each other (maximum coherence) and 0 if they point in opposite directions (minimum coherence):

$$r = \text{coherence}(\theta) = \frac{|\sum_W \mathbf{d}|}{\sum_W |\mathbf{d}|}. \quad (3.4)$$

An example of a local coherence map computed with Eq. (3.4) is shown in Fig. 3.14c. For the gradient-based approach corresponding to Eq. (3.3), it can be shown that Eq. (3.4) simplifies to

$$r_{ij} = \text{coherence}(\theta_{ij}) = \frac{\sqrt{(G_{xx} - G_{yy})^2 + 4G_{xy}^2}}{G_{xx} + G_{yy}}. \quad (3.5)$$

Jain et al. (1997) computed the concordance of the orientations according to their variance in 5×5 neighborhoods whereas Donahue and Rokhlin (1993) computed this according to the residual of the least-square minimization.

The major flaw of gradient-based orientation estimators is their failure in the near-zero gradient regions, namely ridge peaks and valley bottoms. In fact, in these regions, the small values of both the x - and y -components of the gradient imply high noise sensitivity. For this reason, some authors recommend to look beyond the first-order derivatives; see Larkin (2005) for a comprehensive review. Using second-order derivatives only partially solves the problem since the high noise sensitivity is moved to the zero crossing regions (i.e., inflection points) where all the second-order derivatives and the Hessian are null. The method by Da Costa et al. (2001) is based on both first- and second-order derivatives: for each region, a binary decision on which operators to use is taken according to the local coherence of the two operators.

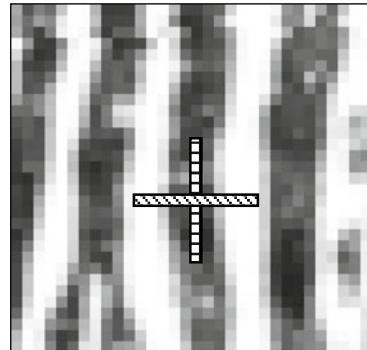
3.3.2 Slit- and Projection-Based Approaches

The first slit-based approach dates back to the 1960s, but some variants have been recently introduced. The basic idea is to define a fixed number (n_S) of reference orientations or slits S_k :

$$S_k = k \frac{\pi}{n_S}, \quad k = 0 \dots n_S - 1$$

and to select the best slit $S_{k_{opt}}$ based on the pixel gray-values along the slits. The local orientation at $[x_i, y_j]$ is the orientation $\theta_{ij} = S_{k_{opt}}$ computed in a local window W centered at $[x_i, y_j]$.

Fig. 3.15 A pair of orthogonal slits with high standard deviation contrast is shown for a local region of a fingerprint



Stock and Swonger (1969) sum the pixel gray values along eight slits and select the minimum-sum slit or maximum-sum slit for ridge or valley pixels, respectively: in fact, for pixels lying on ridges (dark), the sum of gray values along the ridge orientation is small, whereas for pixel lying on valleys (bright), the sum is high.

Based on the observation that the total fluctuation of the gray-scale is the smallest along the orientation of the ridges and largest in the orthogonal direction, similar methods have been proposed by Mehtre et al. (1987), He et al. (2003), and Oliveira and Leite (2008). In particular, Oliveira and Leite (2008) compute the standard deviation $stdev(S_k)$ of the gray-scale of the pixels corresponding to each slit S_k and select the optimal slit according to the maximum standard deviation contrast between a slit and its orthogonal slit (see Fig. 3.15).

Sherlock (2004) suggests projecting the ridge lines inside a local window along a number of discrete orientations (see Fig. 3.21 for an example of gray-scale projection): the projection that exhibits the smallest variation corresponds to the orientation of the ridges within the window. In the Ji and Yi (2008) approach, all the ridge lines except the central one are removed from the local window before computing the projections.

The computational complexity of slit- and projection-based approaches is usually higher than gradient-based techniques, and quantization might produce a coarser angular resolution. However, these methods allow to assign a probability value to each quantized orientation that can be useful to further process noisy regions. In other words, a gradient-based technique leads to a winner-take-all decision where just the optimal orientation is carried over, whereas in slit- and projection-based methods, one can also exploit the probability of the non-winning orientations for subsequent regularization or post-processing.

3.3.3 Orientation Estimation in the Frequency Domain

The Kamei and Mizoguchi (1995) method (well described in Kamei, 2004) is based on the application of 16 directional filters in the frequency domain. The optimal orientation at each pixel is then chosen not only according to the highest filter response but also taking local smoothing into consideration. Analogous results can be achieved in the spatial domain by using the Gabor-like filters, as proposed by Hong et al. (1996) and Nakamura et al. (2004).

The Chikkerur et al. (2007) approach is based on Short-Time Fourier Transform (STFT) analysis. The image is divided into partially overlapped blocks whose intensity values are cosine tapered moving from the center toward the borders. For each block, the Fourier Transform $F(u, v)$ is computed and its spectrum $|F(u, v)|$ is mapped to polar coordinates $|F(r, \theta)|$. The probability of a given θ value (within the block) is then computed as the marginal density function:

$$p(\theta) = \int_r p(r, \theta) dr, \quad \text{where } p(r, \theta) = \frac{|F(r, \theta)|}{\iint_{r, \theta} |F(r, \theta)| dr d\theta}.$$

The expected value of θ for the block is finally estimated, according to Eq. (3.2), as

$$E\{\theta\} = 90^\circ + \frac{1}{2} \text{atan2}\left(\int_0 p(\theta) \sin(2\theta) d\theta, \int_0 p(\theta) \cos(2\theta) d\theta\right). \quad (3.6)$$

Larkin (2005) proposed two energy-based operators that provide uniform and scale-invariant orientation estimation. The second operator, the most robust one, is based on spiral phase quadrature (or the Rietz transform). Although both the operators can be applied also in the spatial domain through convolution, the most natural and simpler implementation of these operators is in the frequency domain.

3.3.4 Orientation Image Regularization

The orientation image \mathbf{D} , computed from poor-quality fingerprints, may contain several unreliable elements due to creases, local scratches, or cluttered noise. In this situation, a local smoothing can be very useful in enhancing \mathbf{D} . This can be done by (re)converting the angles in orientation vectors \mathbf{d} (Eq. (3.1)) and by averaging them through Eq. (3.2). Figure 3.16 shows an example of orientation image smoothing. However, such a simple averaging has some limitations (Fig. 3.16b):

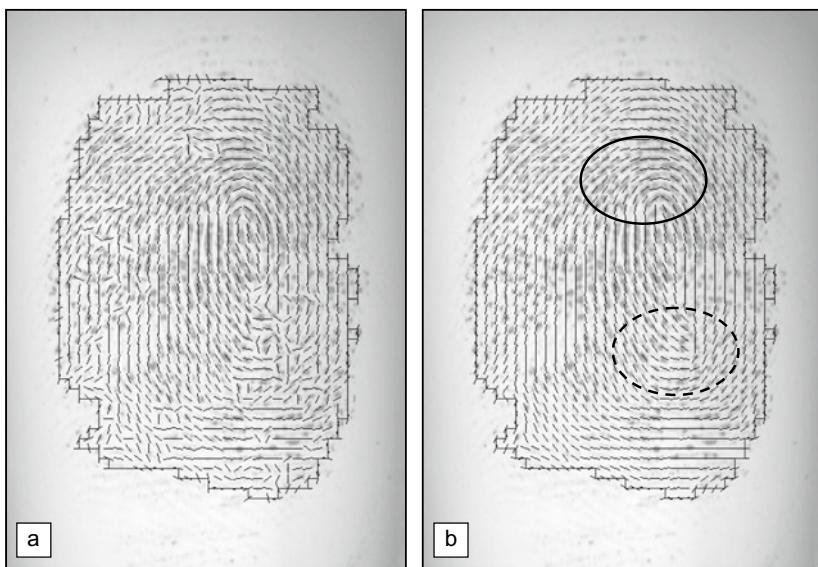


Fig. 3.16 **a** Estimation of local ridge orientation in a fingerprint through the gradient-based approach corresponding to Eq. (3.3): in the noisy regions, the estimate is unreliable; **b** two iterations of local (3×3) smoothing are applied, resulting in a more consistent representation; it is worth noting that while the smoothing recovered the correct orientation at several places (e.g., inside the solid circle), it altered the average orientation inside the region denoted by the dashed circle where incorrect orientations were dominating the correct one

1. It is ineffective when the incorrect orientations dominate the correct ones.
2. Smooths out high curvature values, especially in singular point regions.
3. Slightly shifts the loop singularities.

To overcome the undesired effects described above, more elaborate approaches than a simple average have been proposed:

- Jiang et al. (2004) noted that when a noisy region is smoothed, the local coherence tends to increase. On the other hand, if a high curvature region is smoothed, the local coherence remains low. To prevent smoothing out high curvature regions, the size of the smoothing window is then chosen according to a hierarchical coherence analysis. A similar approach was proposed by Can and Lin (2009).
- Liu et al. (2004) also argued that since the noise is often caused by scars and breaks in ridges, it can be modeled as an impulse function. To suppress such kind of noise, a simple averaging (i.e., a linear filtering) is not effective and a non-linear approach, similar to a median filtering, performs better.

- Zhu et al. (2006) trained a neural network to classify orientation image elements into two classes: correct and incorrect, based on an 11-dimensional feature vector. For each element, the 11 features, including gradient magnitude, gray-scale variance, gray-scale variance projected along the ridge orientation, inter-ridge distance, and variance of the peak/valley heights, are extracted from the corresponding image block. After classification, the incorrect ridge orientations are corrected using the orientation of the neighboring elements.
- Zhang and Yan (2007) define “invalid regions” in the foreground as the sets of connected elements with low coherence value and use the contours of these regions to build a constrained Delaunay triangulation that is used to correct the orientations through interpolation.
- In Oliveira and Leite (2008), correction is based on multi-scale analysis. In particular, they compute the orientation image at two different scales (fine scale and coarse scale) and correct only the elements whose value substantially differs between the two representations; in case of no substantial difference, the fine scale value is retained; otherwise, the coarse scale value is used to correct the fine scale orientation image. Another multi-resolution approach was introduced by Mei et al. (2009).
- Some authors proposed to regularize orientations through diffusions models (Perona, 1998): Hou and Yau (2010) variational approach tries to simultaneously smooth the orientations and preserve the singularity positions; Cao et al. (2012) model considers the distribution of divergence and curl of the orientation vector field. In general, these techniques require to solve non-linear differential equations through iterative numerical methods.

3.3.5 Global Models of Ridge Orientations

A global “mathematical” model for ridge orientation can be very useful for several purposes such as orientation image correction, fingerprint data compression, and synthetic fingerprint generation (see Chap. 7). Hereafter, we make a distinction among the models defined according to the singularities (type and position) from those independent of the singularities.

Models Based on Singularities

Sherlock and Monro (1993) proposed a mathematical model to synthesize a fingerprint orientation image from the position of loops and deltas alone. This approach is also known as the zero–pole model since it takes a loop as a zero and a delta as a pole in the complex plane. But, the model makes some simplifying assumptions and it does not cover all possible fingerprint patterns (fingerprints with different ridge patterns may present the same singularities at the same locations). Improvements to this method have

been proposed by Vizcaya and Gerhardt (1996) and Araque et al. (2002). In particular, Vizcaya and Gerhardt (1996) improved the zero–pole model by using a piece-wise linear approximation around singularities to adjust the zero and pole’s behavior. These new models introduce additional degrees of freedom to better cope with fingerprint pattern variability. Vizcaya and Gerhardt (1996) also proposed an optimization technique, based on gradient descent, to determine the model parameters starting from an estimation of the orientation image. More details on the Sherlock and Monro (1993) and the Vizcaya and Gerhardt (1996) methods can be found in Sect. 7.2.2 where they are used for the synthetic generation of local orientation images.

In the above-mentioned methods, the influence of a singularity on the orientation of a given point does not depend on the distance of the point from the singularity; this can cause errors in regions far from singular points. Furthermore, these models cannot deal with fingerprints with no singularities such as Arch-type fingerprints. Zhou and Gu (2004b) developed a rational complex model, which generalizes the zero–pole model by adding some pseudozeros and pseudopoles as the control points: the pseudozeros are the roots of the additional polynomial in the numerator and the pseudopoles are the roots of the additional polynomial in the denominator. The model parameters can be derived by Weighted Least Square optimization (WLS) starting from the position of the singularities and estimation of the orientation image. Further variants were proposed by Zhou and Gu (2004a) and Gu et al. (2006).

Li et al. (2006) argued that a good orientation model should not only be accurate in approximating the underlying orientation image, but also should have prediction capability where the ridge information is not available (e.g., due to excessive noise). The method proposed by Li et al. (2006) uses a first-order phase portrait approach to compute the predicted orientation. This allows a reconstruction of the orientation using the data around the singular points. To increase the prediction accuracy, the initial estimation of the orientation fields (computed through a gradient-based approach) is refined by replacing the unreliable orientations with the predicted orientations. The refined orientation is then used to obtain the final orientation model using a constrained non-linear phase portrait approach.

Huckemann et al. (2008) argued that most of the global models proposed after Sherlock and Monro’s model are controlled by too many parameters, making it critical to extract stable parameter values from a given orientation image. Their model, denoted as Quadratic Differential (QD), extends the basic Sherlock and Monro (1993) zero–pole model, by using as control parameters (besides the singularity positions) five values with clear geometric meaning: three parameters control the finger placement with respect to the image (i.e., origin and angle of the reference axes) and the remaining two parameters define the horizontal and vertical size of the finger. An extended version of the above method, named XQD, was proposed by Gottschlich et al. (2017).

All the above-mentioned modeling methods rely on the position of singularities and the underlying optimization techniques, but provide no guidance on whether to first compute

the singularity positions based on a noisy orientation map or improve the orientations based on inaccurate singularities. An elegant approach to overcome the above problem was proposed by Dass (2004) whose Bayesian formulation allows the simultaneous extraction of orientation image and singularities. The orientation image is iteratively updated by taking into account the spatial smoothness in a local neighborhood, the gradient values, and contributions to the local orientation given by the singularities. Obtaining orientation image and singularity information simultaneously offers the additional advantage of interleaved updating: the orientation image can be dynamically improved based on current singularity information and the singularity extraction can be performed with higher accuracy thanks to the improved orientation image. Unfortunately, the numerical optimization underlying the joint estimation can be very time-consuming.

Models Independent of the Singularities

Ridge orientations are modeled based on orthogonal polynomials which are defined independently of the singularities. Here, the model coefficients are determined as a data fitting problem starting from an initial coarse estimation:

- Wang et al. (2007a) proposed a fingerprint orientation model based on 2D Fourier series expansion (FOMFE) in the phase plane. The model can seamlessly summarize global features, including high curvature regions around singularities. The extended version by Wang and Hu (2011) can reconstruct the global orientation map starting from a partial fingerprint. Further variants (Tashk et al., 2009; Tao et al., 2010) were introduced to improve the quality of reconstruction in areas of high curvature and poor quality.
- The Ram et al. (2010) model uses the Legendre polynomials to independently model sine and cosine components of the orientation map. In the variant proposed by Jirachaweng et al. (2011), the regions where the basic model does not fit well the data (e.g., around singularities) are refined by using the higher order Legendre polynomials.
- Discrete Cosine Transform (DCT) was used as a basis function by Liu and Liu (2012) and Liu et al. (2014). In the former approach, the orientation field is reconstructed by a linear combination of DCT atoms, and sparse coding is used to improve robustness against noise.
- Bian et al. (2014) proposed a method for reconstructing the orientation field by using the best quadratic approximation by orthogonal polynomials in two discrete variables in the sine domain. The proposed basis functions are true orthogonal in the case of the finite discrete data (unlike for other polynomials such as the Legendre) and the underlying optimization is claimed to be numerically more stable. An extension of this method combined with a final orientation diffusion is the basis of the approach described in Bian et al. (2017a).

Some researchers argued that modeling orientations based on orthogonal polynomials are closer to an approximation method rather than statistical modeling, since the natural variability of fingerprint patterns is not encoded in the model, and, when a large region is dominated by the noise, the recovery ability is limited. Learning-based methods discussed in the following subsection were conceived to overcome this limit.

3.3.6 Learning-Based Methods

One of the first learning-based models was introduced by Lee and Prabhakar (2008) whose approach computes the orientation image based on an MRF (Markov Random Field) made up of two components; one incorporates a global mixture model of orientation fields learned from training fingerprint examples and the other enforces a smoothness constraint over the orientation image in the neighboring regions. Although Lee and Prabhakar (2008) implementation is computationally intensive, it demonstrates the effectiveness of model-based estimation techniques.

Inspired by spelling correction techniques used in natural language processing, Feng et al. (2013) proposed an effective method to exploit the prior knowledge of fingerprint orientations (see Fig. 3.17). A dictionary of reference orientation patches is built off-line using a set of true orientation fields. During the online phase, a first estimation is achieved through one of the known techniques (e.g., STFT), then the orientation map is divided into overlapping patches, and for each patch, a dictionary lookup retrieved the six nearest neighbors dictionary “words”. Finally, an energy minimization process, solved by loopy

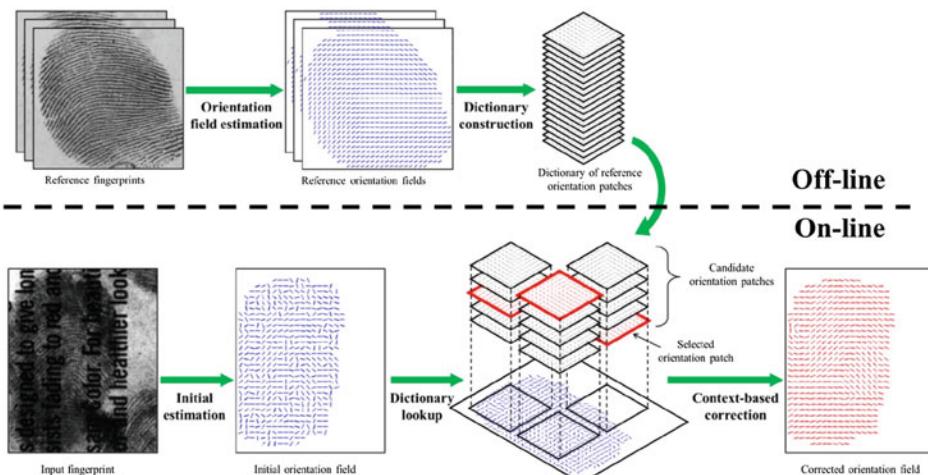
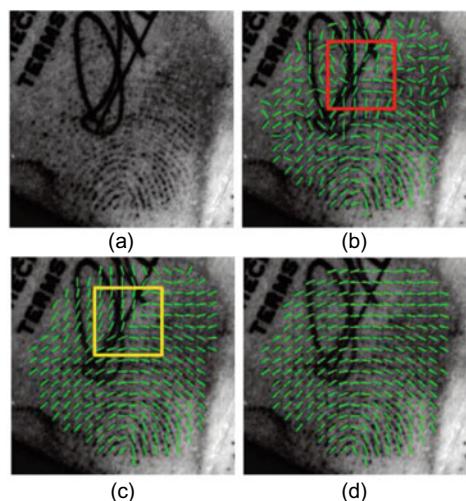


Fig. 3.17 The global-dictionary approach proposed by Feng et al. (2013). © IEEE. Reprinted, with permission, from Feng et al. (2013)

Fig. 3.18 **a** A latent fingerprint; **b** initial orientation extraction by STFT method; **c** global-dictionary approach correction is made through a dictionary patch which is feasible in general but unrealistic at that specific position; and **d** the local-dictionary approach properly handles this case. © IEEE. Reprinted, with permission, from Yang et al. (2014)



belief propagation, is applied to smooth out inconsistencies across neighboring patches when selecting the optimal assignments. Variants of this approach have been introduced by Chen et al. (2016), and Liu and Yang (2017), based on multi-scale analysis and sparse coding, respectively.

Yang et al. (2014) observed that in noisy images, some dictionary patches could be selected at places which are unrealistic given the overall structure of fingerprints (see Fig. 3.18) and proposed to use localized dictionaries for different regions. However, this approach requires fingerprints to be coarsely pre-aligned, which is a critical step for poor-quality and partial images. Therefore, a robust alignment was introduced based on probabilistic voting (i.e., the Hough transform) of all local orientation patches. The local-dictionary approach outperformed the global one in a number of experiments carried out by the authors.

In the method introduced by Cao et al. (2014), the dictionary words are not orientation patches but ridge patches (see Fig. 3.19a). This allows to simultaneously estimate local orientations and frequencies in order to properly apply contextual enhancements (Sect. 3.6.2).

Deep learning techniques based on CNN were proposed by Cao and Jain (2015), and Schuch et al. (2017a); training and inference are quite different in the two approaches:

- In the method by Cao and Jain (2015), (i) a dictionary of 128 orientation patches (of size 10×10) is created by unsupervised clustering good-quality orientation patches extracted from NIST SD4 (Watson & Wilson, 1992); (ii) a training set of 1.28 million good-quality image patches (of size 160×160) is then built from NIST SD14 (Watson,

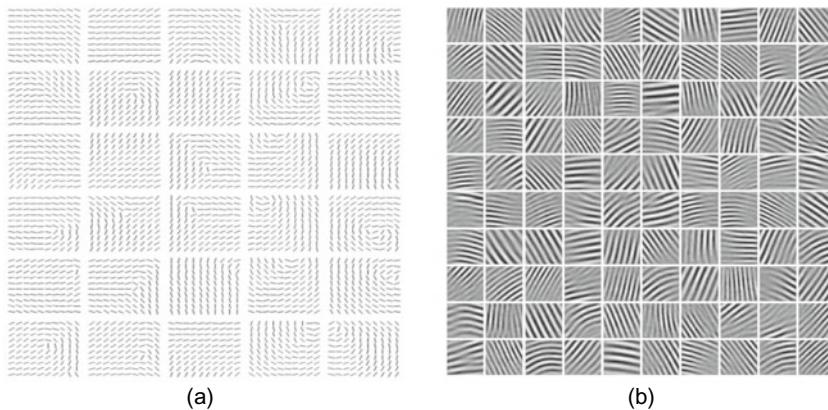


Fig. 3.19 **a** Orientation patches versus **b** ridge patches. © IEEE. Reprinted, with permission, from Cao et al. (2014)

1993) by labeling each image patch with the index of the closest dictionary word¹; (iii) the training set patches are corrupted by line-like texture noise and block cancellation and used to train a CNN to classify an input patch into one of the 128 classes; (iv) during inference, the input image is divided into overlapping patches and each of them is classified by the CNN; and (v) the final orientation of each element in the input is determined by fusing the orientations of the covering patches.

- In the CNN approach by Schuch et al. (2017a), training is supervised and based on the manually marked orientations made available in the FVC-onGoing FOE training set (see Sect. 3.3.7). This approach is closer to a typical CNN-based semantic segmentation (Minaee et al., 2021) where a fully convolutional architecture is exploited to simultaneously process the whole image (with no need of sliding a window over local patches). The hierarchical arrangement of convolutional layers increases the receptive field size as one moves from the input to the output level, so that the prediction of each output element depends on a sufficiently large image portion. The authors empirically show that a classification network with a soft fusion of output labels (i.e., deep expectation) outperforms a similar model trained for orientation regression.

3.3.7 Benchmarking Fingerprint Orientation Extraction

While many researchers evaluated the efficacy of their orientation extraction approach on indirect tasks (e.g., the improvement of singularity detection or fingerprint recognition),

¹ The similarity between an orientation patch and an image patch is computed by extracting the orientations from the image patch (with a local method) and using a squared cosine distance.

Table 3.1 A comparison of known approaches on FVC-onGoing FOE. The average error on the bad-quality partition (AvgError Bad) is the main indicator

Method	AvgError bad (°)	AvgError good (°)	Computation time on Intel Xeon E5410 (s)
CNN DEX-OF (Schuch et al., 2017a)	6.73	4.84	1.776
Local dictionary (Yang et al., 2014)	9.66	6.08	5.987
ROF (Div-Curl model) (Cao et al., 2012)	11.20	5.24	0.762
Adaptive polynomials (Turroni et al., 2011)	13.27	5.93	4.772
FOMFE (Wang et al., 2007a)	21.44	6.70	1.996
Gradient (baseline) (Turroni et al., 2011)	21.83	5.86	0.074

we believe that assessing this important module in isolation is very valuable because it can provide useful insights on the residual errors and guide future improvements.

FVC-onGoing FOE (Fingerprint Orientation Extraction) is a benchmark² specifically designed to evaluate the accuracy of orientation extraction algorithms. Even if the fingerprints used in FVC-onGoing FOE are plain impressions, methods conceived to work on latent prints can be applied to this dataset as well since recovering orientations on noisy latent fingerprints is not much different than processing poor-quality plain impressions. Manual labeling of orientations at selected places and subsequent interpolation is used to create a ground truth for FVC-onGoing FOE: even if this procedure is prone to inaccuracies (as pointed out by Schuch et al., 2017b), looking at the orientation error on the poor-quality partition of this benchmark gives a better overview of the state of the art (see Table 3.1). From Table 3.1, the advantages of learning-based techniques (Local Dictionary and CNN) on poor-quality images are evident and their robustness is extremely useful for latent fingerprint processing; however, their computational complexity is typically much higher than local techniques such as the gradient baseline, whose accuracy on reasonable-quality samples is comparable and more than satisfactory for several applications.

Further comparisons of fingerprint orientation techniques can be found in Turroni et al. (2011) and in the recent survey by Bian et al. (2019).

² <https://www.biolab.csr.unibo.it/fvcongoing/UI/Form/BenchmarkAreas/BenchmarkAreaFOE.aspx>.

3.4 Local Ridge Frequency Estimation

The local ridge frequency (or density) f_{xy} at point $[x, y]$ is the number of ridges per unit length along a hypothetical segment centered at $[x, y]$ and orthogonal to the local ridge orientation θ_{xy} . A frequency image \mathbf{F} , analogous to the orientation image \mathbf{D} , can be defined if the frequency is estimated at discrete positions and arranged into a matrix.

The local ridge frequency varies across different fingers and may also noticeably vary across different regions of the same fingerprint (see Fig. 3.20). Orczyk and Wieclaw (2011), based on manual marking by a forensic expert, reported that for 500 dpi fingerprints, most local ridge distances lie in the range 5–10 pixels with an average of 7.71.

Hong et al. (1998) estimate local ridge frequency by counting the average number of pixels between two consecutive peaks of gray levels along the direction normal to the local ridge orientation (see Fig. 3.21). For this purpose, the surface S corresponding to the fingerprint is sectioned with a plane parallel to the z -axis (see Fig. 3.9) and orthogonal to local ridge orientation. The frequency f_{ij} at $[x_i, y_j]$ is computed as follows.

1. A 32×16 oriented window centered at $[x_i, y_j]$ is defined in the ridge coordinate system (i.e., rotated to align the y -axis with the local ridge orientation).
2. The x -signature of the gray levels is obtained by accumulating, for each column x , the gray levels of the corresponding pixels in the oriented window. This is a sort of

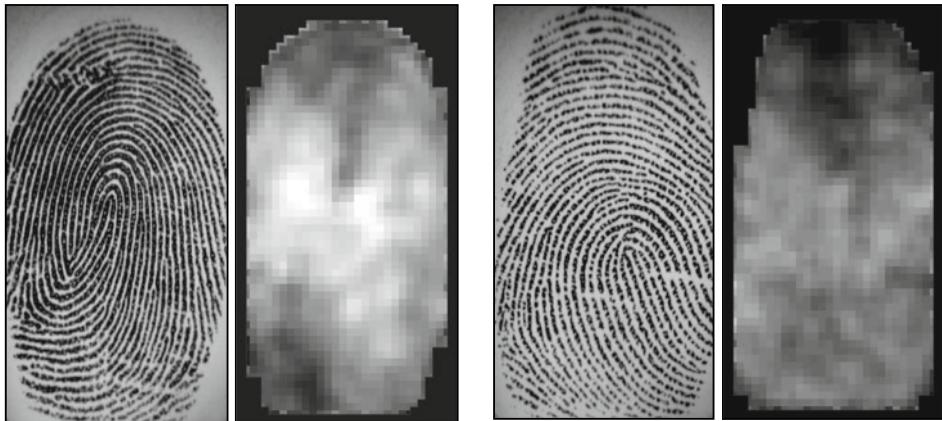


Fig. 3.20 Two fingerprint images and the corresponding frequency image computed with the method proposed by Maio and Maltoni (1998a). A local 3×3 averaging is performed after frequency estimation to reduce noise. Light blocks denote higher frequencies. It is quite evident that significant changes may characterize different fingerprint regions and different average frequencies may result from different fingers

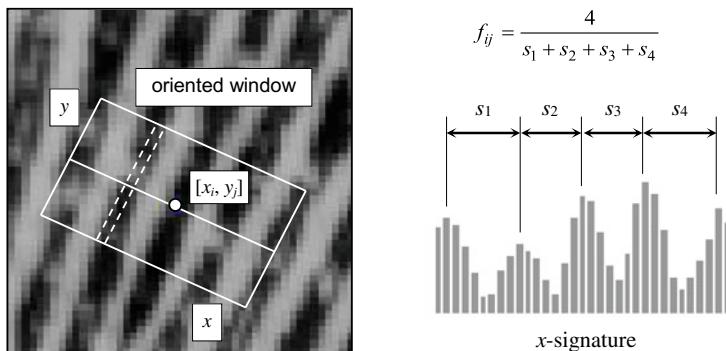


Fig. 3.21 An oriented window centered at $[x_i, y_j]$; the dashed lines show the pixels whose gray levels are accumulated for a given column of the x -signature (Hong et al., 1998). The x -signature on the right clearly exhibits five peaks; the four distances between consecutive peaks are averaged to determine the local ridge frequency

averaging that makes the gray-level profile smoother and prevents ridge peaks from being obscured due to small ridge breaks or pores.

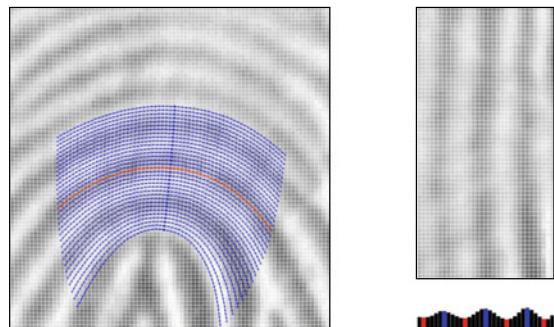
3. f_{ij} is determined as the inverse of the average distance between two consecutive peaks of the x -signature.

The method is simple and fast. However, it is difficult to reliably detect consecutive peaks of gray levels in the spatial domain in noisy fingerprint images. In this case, the authors suggest using interpolation and low-pass filtering. An alternative way to extract ridge distances from the x -signature makes use of a fitting method based on the first- and second-order derivatives (Yang et al., 2003).

Jiang (2000) and Orczyk and Wieclaw (2011) also compute the local ridge frequency starting from the x -signatures. Jiang (2000), instead of measuring the distances in the spatial domain, makes use of a high-order spectrum technique called *mix-spectrum*. The ridge patterns in a fingerprint image are noisy periodic signals; when they deviate from a pure sinusoid shape, their energy is distributed to their fundamental frequency and harmonics. The mix-spectrum technique enhances the fundamental frequency of the signal by exploiting the information contained in the second and third harmonic.

Gottschlich (2012) argued that most of the frequency estimation errors of x -signature-based methods occur in curved regions (e.g., close to singularities) where a single dominant orientation cannot be defined. Therefore, he proposed to locally rectify ridges based on ridge flow curvature before computing the x -signature (Fig. 3.22). The rectification also allows to increase the window size.

Fig. 3.22 The curvature-based rectification of ridges proposed by Gottschlich (2012) to improve *x-signature*-based local frequency estimation. © IEEE. Reprinted, with permission, from Gottschlich (2012)



In the method proposed by Maio and Maltoni (1998a), the ridge pattern is locally modeled as a sinusoidal-shaped surface, and the variation theorem is exploited to estimate the unknown frequency. The variation V of a function h in the interval $[x_1, x_2]$ is the amount of “vertical” change in h :

$$V(h) = \int_{x_1}^{x_2} \left| \frac{dh(x)}{dx} \right| \cdot dx.$$

If the function h is periodic at $[x_1, x_2]$ or the amplitude changes within the interval $[x_1, x_2]$ are small, the variation may be expressed as a function of the average amplitude α_m and the average frequency f (see Fig. 3.23).

$$V(h) = (x_2 - x_1) \cdot 2\alpha_m \cdot f.$$

Therefore, the unknown frequency can be estimated as

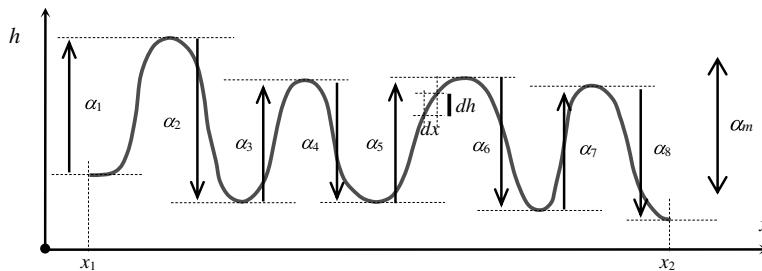


Fig. 3.23 The variation of the function h in the interval $[x_1, x_2]$ is the sum of amplitudes $\alpha_1, \alpha_2, \dots, \alpha_8$ (Maio & Maltoni, 1998a). If the function is periodic or the function amplitude does not change significantly within the interval of interest, the average amplitude α_m can be used to approximate the individual α values. Then the variation may be expressed as $2\alpha_m$ multiplied by the number of periods of the function over the interval

$$f = \frac{V(h)}{2 \cdot (x_2 - x_1) \cdot \alpha_m}. \quad (3.7)$$

Maio and Maltoni (1998a) proposed a practical method based on the above analysis. The variation and the average amplitude of a two-dimensional ridge pattern are estimated from the first- and second-order partial derivatives and the local ridge frequency is computed from Eq. (3.7). Two examples of frequency images computed using this method are shown in Fig. 3.20.

Kovacs-Vajna et al. (2000) proposed a two-step procedure: first, the average ridge distance is estimated for each 64×64 sub-block of the image that is of sufficient quality and then this information is propagated, according to a diffusion equation, to the remaining regions. Two methods are considered in the first step: geometric and spectral. In the geometric approach, the central points of the ridges are computed on a regular grid and the ridge distances are measured on straight lines passing through these points. Unlike the x -signature approach, distances are directly measured in the two-dimensional image; several estimates on the same image block are performed to compensate for the noise. The second method is based on a search of the maxima in the Fourier power spectrum of each sub-block. Here too, the method works on two-dimensional signals. The invariance with respect to the local ridge orientations is obtained by performing the maxima search in radial directions: in fact, all the components (harmonics) having the same distance from the origin denote the same frequency.

Another geometrical method was introduced by Lehtihet et al. (2014); gray-level minima (lying on ridges) are used as the vertex for a Delaunay triangulation and the resulting graph is locally parsed to estimate distances of adjacent ridges.

Almansa and Lindeberg (1997, 2000) use scale space theory to locally estimate ridge width; their method relies upon combinations of normalized derivatives computed pointwise.

The approach by Chikkerur et al. (2007), already discussed in Sect. 3.3.3 for the estimation of the local ridge orientations, is based on Short-Time Fourier Transform (STFT) analysis. The probability of a given r value (corresponding to the average frequency f within the block) is computed as the marginal density function:

$$p(r) = \int_{\theta} p(r, \theta) d\theta, \quad \text{where } p(r, \theta) = \frac{|F(r, \theta)|}{\iint_{r, \theta} |F(r, \theta)| dr d\theta}$$

and the expected value of r for the block is estimated as

$$E\{r\} = \int_r p(r) r dr. \quad (3.8)$$

Zhan et al. (2006) compared frequency estimation approaches operating in the spatial domain versus Fourier domain and concluded that the former can be implemented more efficiently but the latter seems to be more robust to noise.

3.5 Singularity Detection and Pose Estimation

Singularities are the most evident Level-1 features in the fingerprint pattern. They can be useful for coarse fingerprint classification (Chap. 5) and to define stable points for pose estimation and absolute registration. Most of the approaches proposed in the literature for singularity detection operate on the fingerprint orientation image. In the rest of this section, the main approaches are presented. These include algorithms based on the Poincaré index, local characteristics of the orientation image, partition of the orientation image, and global model of the orientation image. Each of these four families of algorithms is described in Sects. 3.5.1–3.5.4.

3.5.1 Poincaré

An elegant and practical method based on the Poincaré index was proposed by Kawagoe and Tojo (1984). Let \mathbf{G} be a vector field and C be a curve immersed in \mathbf{G} ; then the Poincaré index $P_{\mathbf{G},C}$ is defined as the total rotation of the vectors of \mathbf{G} along C (see Fig. 3.24).

If \mathbf{G} is the discrete vector field associated with a fingerprint orientation image (Note that a fingerprint orientation image is not a true vector field in as much as its elements are unoriented directions (i.e., in the range $[0\dots 180^\circ]$) \mathbf{D} , and $[i,j]$ is the position of the element θ_{ij} in the orientation image, then the Poincaré index $P_{\mathbf{G},C}(i, j)$ at $[i, j]$ is computed as follows.

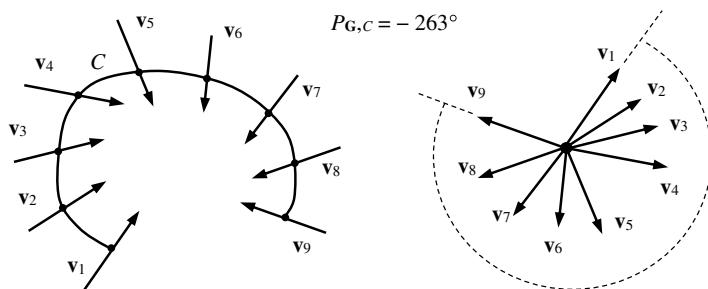


Fig. 3.24 The Poincaré index computed over a curve C immersed in a vector field \mathbf{G}

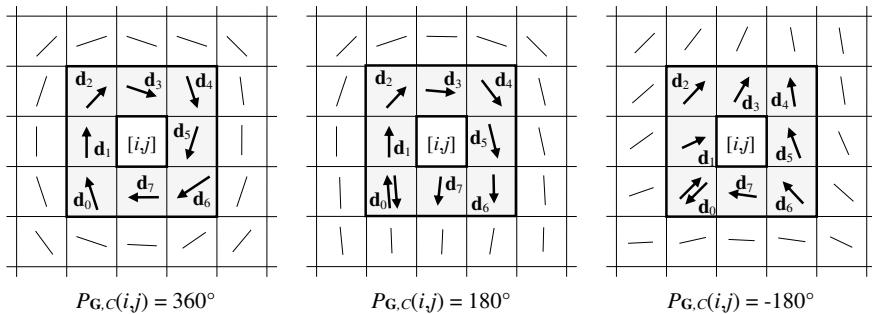


Fig. 3.25 Examples of the Poincaré index computation in the eight-neighborhood of points belonging (from left to right) to a whorl, loop, and delta singularity, respectively. Note that, for the loop and delta examples (center and right), the sense of \mathbf{d}_0 is first chosen upward (to compute the angle between \mathbf{d}_0 and \mathbf{d}_1) and then successively downward (when computing the angle between \mathbf{d}_7 and \mathbf{d}_0)

- The curve C is a closed path defined as an ordered sequence of some elements of \mathbf{D} , such that $[i, j]$ is an internal point.
- $P_{G,C}(i, j)$ is computed by algebraically summing the orientation differences between the adjacent elements of C . Summing orientation differences requires a *sense*³ (among the two possible values) to be associated at each orientation, in order to transform it into a direction. A solution to this problem is to randomly select the sense of the first element and assign the direction closest to that of the previous element to each successive element. It is well known and can be easily shown that, on closed curves, the Poincaré index assumes only one of the discrete values: $0^\circ, \pm 180^\circ$, and $\pm 360^\circ$. In the case of fingerprint singularities,

$$P_{G,C}(i, j) = \begin{cases} 0^\circ & \text{if } [i, j] \text{ does not belong to any singular region} \\ 360^\circ & \text{if } [i, j] \text{ belongs to a whorl type singular region} \\ 180^\circ & \text{if } [i, j] \text{ belongs to a loop type singular region} \\ -180^\circ & \text{if } [i, j] \text{ belongs to a delta type singular region} \end{cases}$$

Figure 3.25 shows three portions of the orientation image. The path defining C is the ordered sequence of the eight elements \mathbf{d}_k ($k = 0, \dots, 7$) surrounding $[i, j]$. The direction of the elements \mathbf{d}_k is chosen as follows: \mathbf{d}_0 is directed upward; \mathbf{d}_k ($k = 1, \dots, 7$) is directed so that the absolute value of the angle between \mathbf{d}_k and \mathbf{d}_{k-1} is less than or equal to 90° . The Poincaré index is then computed as

³The sense of a vector is specified by the order of two points on a line parallel to the vector. Orientation and sense together determine the direction of a vector.

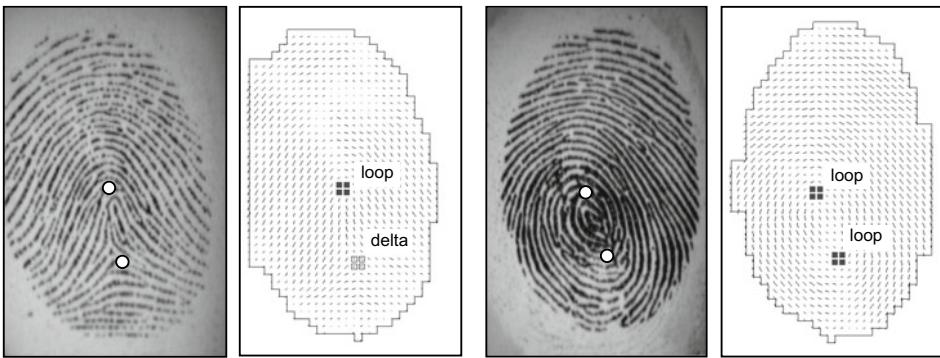


Fig. 3.26 Singularity detection using the Poincaré index for two images. The elements whose Poincaré index is 180° (loop) or -180° (delta) are enclosed by small boxes. Usually, more than one point (four points in these examples) is found for each singular region: hence, the center of each singular region can be defined as the barycenter of the corresponding points. Note that the position of the loop singularities is slightly moved toward the borders because of the local smoothing of the orientation image

$$P_{G,C}(i, j) = \sum_{k=0 \dots 7} \text{angle}(\mathbf{d}_k, \mathbf{d}_{(k+1) \bmod 8}).$$

Some examples of singularities detected by the above method are shown in Fig. 3.26.

An interesting implementation of the Poincaré method for locating singular points was proposed by Bazen and Gerez (2002): according to Green's theorem, a closed line integral over a vector field can be calculated as a surface integral over the rotation of this vector field; in practice, instead of summing angle differences along a closed path, the authors compute the “rotation” of the orientation image (through a further differentiation) and then perform a local integration (sum) in a small neighborhood of each element. Bazen and Gerez (2002) also provided a method for associating an orientation with each singularity; this is done by comparing the orientation image around each detected singular point with the orientation image of an ideal singularity of the same type.

Singularity detection in noisy or low-quality fingerprints is difficult and the Poincaré method may lead to the detection of false singularities (Fig. 3.27). Regularizing the orientation image through a local averaging, as discussed in Sect. 3.3.4, is often quite effective in preventing the detection of false singularities, even if it can lead to the slight displacement of the loop position toward the borders. Wang et al. (2007b) propose a posteriori correction of loop location to compensate for the offset introduced by fingerprint image smoothing.

Based on the observation that only a limited number of singularities can be present in a fingerprint, Karu and Jain (1996) proposed to iteratively smooth the orientation image (through averaging) until a valid number of singularities is detected by the Poincaré index.

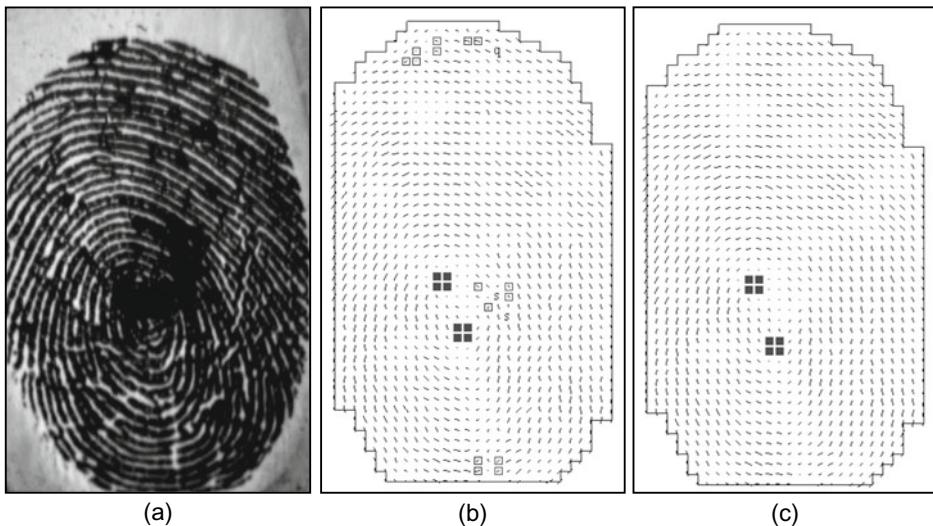


Fig. 3.27 **a** A poor-quality fingerprint; **b** the singularities of the fingerprint in (a) are extracted through the Poincaré method (circles highlight the false singularities); and **c** the orientation image has been regularized and the Poincaré method no longer provides false alarms

In fact, a simple analysis of the different fingerprint classes (refer to Chap. 5 for a more detailed discussion) shows the following:

- Arch fingerprints do not contain singularities.
- Left loop, right loop, and tented arch fingerprints contain one loop and one delta.
- Whorl fingerprints contain two loops (or one whorl) and two deltas.

The above constraints are nicely demonstrated by Zhou et al. (2007) who conclude that for each completely captured fingerprint, the number of loops and deltas are identical. A practical way to enforce this constraint is to compute the Poincaré index along the external boundary of the orientation image and then use the resulting value to limit the number of valid configurations. Another useful suggestion given by Zhou et al. (2007) is to locally change the path C for the computation of the Poincaré index according to the reliability of the underlying elements; in fact, $P_{G,C}$ being independent of the closed path C , if the eight-neighborhood path (shown in Fig. 3.25) includes unreliable elements, C can be progressively enlarged to include more reliable elements. In their approach, Zhou et al. (2009a) use the Poincaré index in conjunction with another similar operator, called DORIC, that looks for the presence of a single peak in the sequence of direction differences $\mathbf{d}_k, \mathbf{d}_{k+1}$; in Fig. 3.25, it can be noted that for loop and delta singularities, there is a single sharp change of direction between \mathbf{d}_7 and \mathbf{d}_0 . After the Poincaré-based detection and DORIC-based filtering, Zhou et al. (2009a) select the optimal subset of

singular points as the set S that minimizes the difference between the estimated orientation image and a reconstruction of the orientation image through the Sherlock and Monroe (1993) model with S as the set of singularities.

Finally, the spatial distributions of singularity locations in nature (Cappelli & Maltoni, 2009) can be exploited to guide singularity extraction or implement post-filtering approaches.

3.5.2 Methods Based on Local Characteristics of the Orientation Image

Some authors have proposed singularity detection approaches where the fundamental idea is to explore the orientation image regions characterized by high irregularity, curvature, or symmetry. In fact, the singularities are the only foreground regions where a dominant orientation does not exist and ridges assume high curvature.

The coherence operator, as defined by Eq. (3.4), was used by Cappelli et al. (1999) to find approximate locations of singular regions. Figure 3.28 shows an example of a coherence map computed over 3×3 neighborhoods.

Srinivasan and Murthy (1992) extract singularities according to the local histogram of the orientation image; in fact, the points where the local histogram does not exhibit a well-pronounced peak are likely to belong to singular regions. By analyzing the predominant orientation in some predefined sectors around the candidate points, Srinivasan and Murthy were able to discriminate between loop and delta singularities.

Chen et al. (2011b) compute local orientation entropy at three different quantization levels and combine these estimates at each position (by simple multiplication) to improve singularity detection robustness against noise.

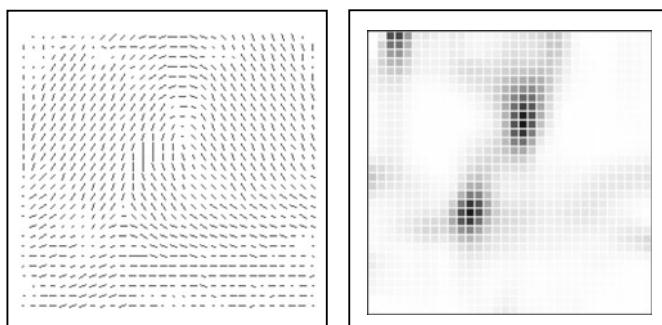


Fig. 3.28 An orientation image and the corresponding coherence map that clearly identifies the regions (dark cells \leftrightarrow low coherence) containing the singularities

The Local Axial Symmetry (LAS) introduced by Liu et al. (2006) is a pixel-wise feature, derived from the orientation image that denotes the symmetry of the orientations inside circular regions centered at each pixel. Based on the observation that there are two regions with lower LAS value on the two opposite sides of each loop-type singularity, a simple binarization scheme is proposed to isolate these two regions and locate the loop(s) as midpoint(s) of their barycenters. A similar symmetry-based technique was used by Liu et al. (2005) to check the validity of singular points (both loop and delta) initially detected through the Poincaré index approach.

Zhu et al. (2016) proposed to derive Walking Directional Fields (WDFs) from the orientation map. Since WDF local vectors point to the singularity centers, a navigation algorithm can be designed to detect singularities without processing the entire orientation map.

Some researchers argued that multi-resolution analysis is an effective tool for singularity detection since it allows (i) to increase robustness against the noise if the same singularities are detected at different resolution; (ii) to locate singular points with sub-block accuracy, and (iii) to reduce computational costs when a coarse-to-fine approach is adopted:

- Koo and Kot (2001) detect the singularities with single pixel accuracy. At each resolution, they derive from the orientation image a curvature image whose blocks indicate the local degree of ridge curvature. High-curvature blocks, which denote singularities, are retained and analyzed at finer resolutions.
- Nilsson and Bigun (2002a, b, 2003) multi-resolution analysis is based on the convolution of the orientation field with two complex filters tuned to detect points of rotational symmetry. The two filters (one for loop-type singularity and one for delta-type singularity) are convolved with orientation image and the points where the response of any one filter is high are retained and analyzed at a finer resolution (see Fig. 3.29). Orientation of the detected singularities can also be estimated from the argument of the complex filter responses. This technique can be implemented in an efficient way thanks to the separability of the filters used. An improved version of this approach is described in Chikkerur and Ratha (2005).
- Liu et al. (2004) approach iteratively checks for the existence of singularities and refines their position over a sequence of resolutions, based on local features such as the Poincaré index and orientation variance.
- Wang et al. (2007b) multi-resolution approach is based on harmonic relationships between the orientation of a singular point and its neighbors.
- In the approach by Weng et al. (2011), the candidate singularities detected by an improved Poincaré index method at different resolutions (varying block size and stride) are aggregated and post-filtered according to the orientation coherence.

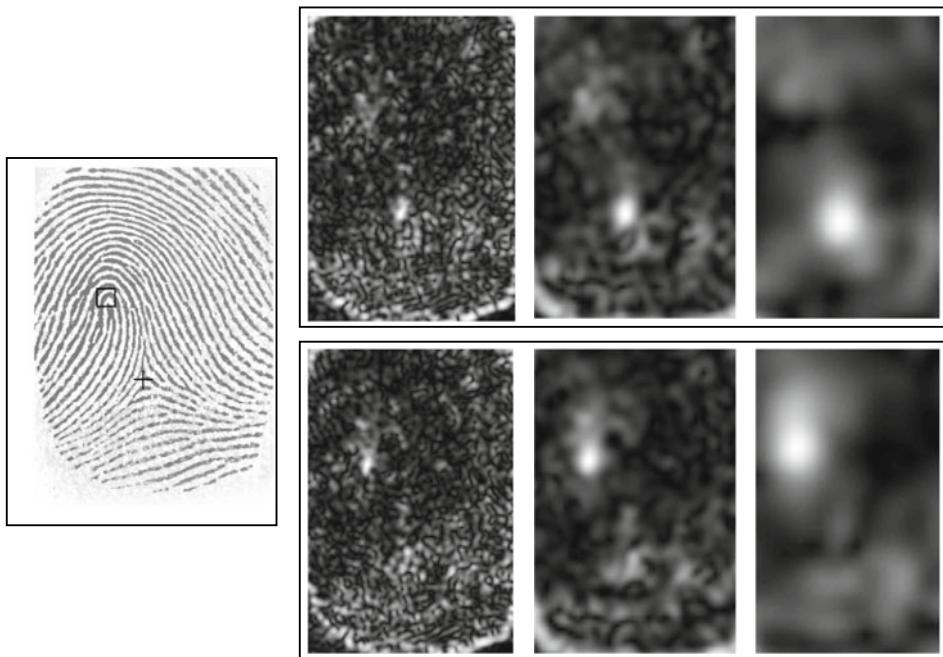
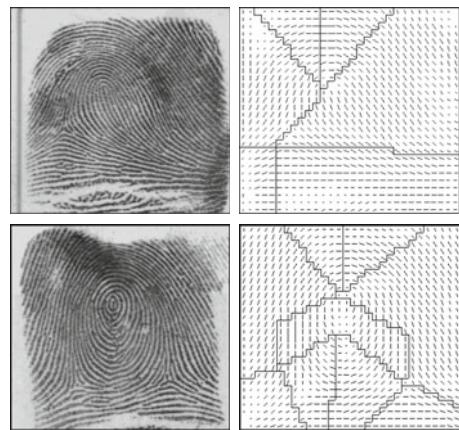


Fig. 3.29 An example of the Nilsson and Bigun (2003) approach. A fingerprint image and its responses of the loop filter (bottom row) and delta filter (top row) at three different scales are shown. Images courtesy of J. Bigun

3.5.3 Partitioning-Based Methods

Some authors noted that partitioning the orientation image in regions characterized by homogeneous orientation implicitly reveals the position of singularities. Hung and Huang (1996) and Huang et al. (2007) coarsely discretize the orientation image by using a very small number of orientation values. Each orientation value determines a region. The borderline between two adjacent regions is called a fault line. By noting that fault lines converge toward loop singularities and diverge from deltas, the authors define a geometrical method for determining the convergence and divergence points. In Maio and Maltoni (1996) and Cappelli et al. (1999), the orientation image is partitioned by using an iterative clustering algorithm and a set of dynamic masks, respectively (Fig. 3.30). Ohtsuka and Takahashi (2005), Ohtsuka and Kondo (2005), and Ohtsuka and Watanabe (2010) derive the position of singularities from the extended relational graphs modeling the orientation image. Rämö et al. (2001) and Kryszeck and Drygajlo (2006) implicitly partition the orientation image in correspondence with the points where the x - and y -orientation components (refer to Eq. (3.1)) change sign; efficient methods are then introduced to extract singularities based on the contemporary zero crossings of both orientation components.

Fig. 3.30 Orientation image partitioning with the MASK approach (Cappelli et al., 1999). The intersections between region border lines denote fingerprint singularities. © IEEE. Reprinted, with permission, from Cappelli et al. (1999)



3.5.4 Methods Based on a Global Model of the Orientation Image

An effective strategy to improve robustness is to exploit a global model of the orientation image (see Sect. 3.3.5):

- Wu and Zhou (2004) used the zero–pole model and the Hough transform to detect the position of singularities starting from the orientation image. In the zero–pole model, the orientation around each singularity is not much influenced by the other singularities, and therefore, within a local window, the singularities’ position and the surrounding orientations are bound by simple linear equations. This makes it possible to implement the voting scheme which is the basis of the Hough transform. An initial location of the singularities is performed with the Poincaré approach; each singularity is then checked and its location refined through the Hough transform.
- Fan et al. (2008) approach is also based on the zero–pole model and the Hough transform. The robustness of this method to identify a single dominant loop (or delta) for a given fingerprint area is demonstrated for real fingerprint images (including noisy samples). However, to refine the position of singularities and to deal with the presence of more loops/deltas, other heuristics based on the Poincaré index and ridge tracing are used in conjunction with the zero–pole model and the Hough transform. See Fig. 3.31 for an example.
- Dass (2004) used the simultaneous computation of orientation image and singularities that is briefly discussed in Sect. 3.3.5.
- Wang et al. (2007a) exploited their global model (FOMFE) to obtain an analytical expression of the local orientation topology: in particular, the classification of each point as {normal point, loop or delta} is determined by the sign of the determinant of

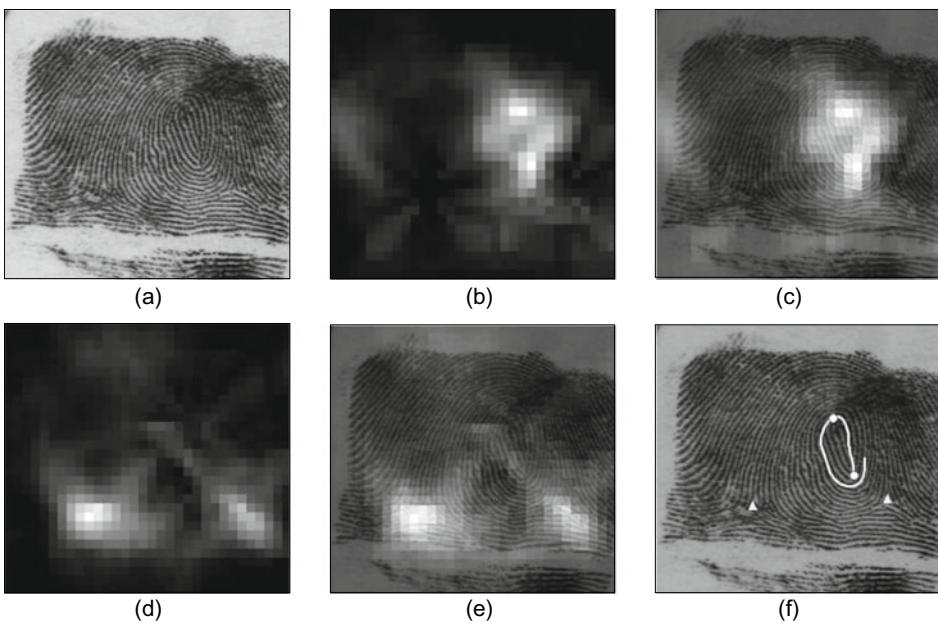


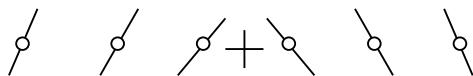
Fig. 3.31 An example of singularity detection with Fan et al. (2008) approach. **a** Original image; **b** Hough space (loop); **c** Hough space (loop) superimposed over the original image; **d** Hough space (delta); **e** Hough space (delta) superimposed over the original image; and **f** detected loops (circle) and deltas (triangle), ridge tracing is also shown. © IEEE. Reprinted, with permission, from Fan et al. (2008)

a 2×2 matrix, called the characteristic matrix. The computation of additional information such as the local curl and divergence assists in implementing effective rules for the removal of false singular points (Wang & Hu, 2008).

3.5.5 Fingerprint Pose Estimation

Fingerprint pose estimation in two dimensions consists in the determination of a (stable) reference point and a direction. The availability of such information allows to shift/rotate fingerprint images in order to achieve an *absolute registration* (or *absolute pre-alignment*). This can be very useful to extract pose independent (fixed-length) features which are needed by many fingerprint indexing methods (Chap. 5) and matching techniques in the encrypted domain (Chap. 9). Note that absolute registration discussed in this section differs from relative registration addressed in Chap. 4; in fact, relative registration is aimed

Fig. 3.32 The core point “+” located on the chosen sextet



at computing the displacement/rotation between a pair of fingerprints while absolute registration operates on a single fingerprint.

Reference Point Detection

The fingerprint core is a natural candidate reference point. Once the singularities have been extracted, the core position may be simply defined as the location of the northernmost loop. There is an ambiguity with the arch-type fingerprints that do not have singularities; for these fingerprints, the estimated flow field is flat and hence no core position can be located. When the core point is detected with the aim of registering fingerprint images (thus obtaining invariance with respect to (x, y) displacement), the core's location may be quite critical and an error at this stage often leads to a failure of subsequent processing steps (e.g., matching). On the other hand, if the core has to be used only for fingerprint registration, it is not important to find the northernmost loop exactly and any stable point in the fingerprint pattern is suitable.

One of the first automated methods for fingerprint registration was proposed by Wegstein (1982). This method, known as R92, searches for a core point independently of the other singularities. The core is searched by scanning (row by row) the orientation image to find *well-formed arches*; a well-formed arch is denoted by a sextet (set of six) of adjacent elements whose orientations comply with predefined rules. One sextet is chosen among the valid sextets by evaluating the orientation of the elements in adjacent rows. The exact core position is then located through interpolation (Fig. 3.32). Even though R92 is quite complicated and heuristic in nature, it usually gives good results and is able to localize the core point with sub-block accuracy. This algorithm was a fundamental component of the fingerprint identification systems used by the FBI.

Several other ideas for the location of stable registration points have been proposed. Novikov and Kot (1998) define the core as the crossing point of the lines normal to the ridges (Fig. 3.33) and used the Hough transform (Ballard, 1981) to determine its

Fig. 3.33 The straight lines normal to the ridges identify a valid registration point that corresponds to the center of curvature



coordinates. Similarly, Rerkrai and Areekul (2000) define the *focal point* as the point where pairs of straight lines normal to the ridges intersect. Because the ridges do not draw perfect concentric circumferences around the core, the normal lines (dashed lines in Fig. 3.33) do not exactly cross at a single point and a sort of average point has to be defined as the center of curvature. Novikov and Kot (1998) compute this average point in the least squares sense, whereas Rerkrai and Areekul (2000) compute the barycenter of the crossing between pairs of normals.

A robust voting schema based on the Hough transform was proposed by Yang et al. (2014). In this approach, votes are cast by individual 4×4 orientation patches. During an off-line training stage, orientation patches are clustered. Then, for each cluster, a statistical model is learnt to denote the relative displacement between the patch and the fingerprint center. During online center computation, each patch is mapped to the most similar cluster and the corresponding probability map is summed to the Hough image. At the end of the voting, the maximum in the Hough image provides a robust estimation of the center position. More details on this method, which was specifically introduced for latent fingerprints, are provided in Sect. 6.4.2. The pose estimation method by Gu et al. (2017) also relies on voting but in their Hough forest-based approach (Gall & Lempitsky, 2013), multiple trees are learnt and their votes are combined to improve robustness.

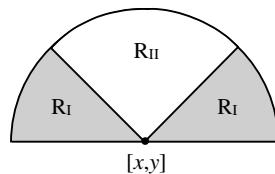
Although the focal point (or the center of curvature) does not necessarily correspond to the core point, it has been experimentally demonstrated to be quite stable with respect to fingerprint variation (displacement, rotation, distortion, etc.). Therefore, it can be reliably used for fingerprint registration. The main problem of these methods is in isolating a fingerprint region characterized by a single center of curvature. In fact, if the selected fingerprint region contains more than one singularity, the result may be unpredictable. To solve this problem, Areekul et al. (2006) proposed an algorithm that separately determines a focal point for each neighbor of a high curvature point and then tries to reach a consensus.

The focal point is also used in the method by Deerada et al. (2020) aimed at determining the pose of latent fingerprints: since orientation estimation in latent fingerprints is not reliable enough to be used for direct focal point computation, the authors proposed a closed-loop approach that progressively locates potential poses and enhances weak friction ridges that form and support these poses through an iterative feedback.

Jain et al. (2000) proposed a multi-resolution approach for locating the northmost loop-type singularities (core) based on the integration of sine components in two adjacent regions R_I and R_{II} (Fig. 3.34). The geometry of the two regions is designed to capture the maximum curvature in concave ridges. At each scale and for each candidate position $[x, y]$, the sine components of the orientation image are integrated over the two regions resulting in the values SR_I and SR_{II} . The points $[x, y]$ that maximize the quantity $(SR_I - SR_{II})$ are retained as candidate positions and analyzed at a finer resolution.

Another interesting multi-resolution approach, designed by Jiang et al. (2004), performs the core localization by means of a hierarchical analysis of orientation coherence:

Fig. 3.34 Regions of integration of the sine components in the method proposed by Jain et al. (2000).



the core is chosen as the point having local minimum coherence at both large and fine scales. This approach was able to correctly locate the core position in about 95% of the FVC2000 DB2 fingerprint images (Maio et al., 2002). An improvement of the above technique, where the orientation coherence is combined with the direction of curvature, was proposed by Van and Le (2009a).

Inspired by the filtering approaches proposed by Nilsson and Bigun (2003) and Chikkerur and Ratha (2005) for singularities detection (see Sect. 3.5.2), Le and Van (2012) designed a new complex filter to detect the core point based on orientation symmetries. Compared to the parabolic symmetry filter of Nilsson and Bigun (2003), the new filter also works with arch fingerprints and is able to deal better with double loop prints. A second operator, based on vertical orientation variation, is used to filter out unwanted singularities (e.g., the southmost loop in a double loop) or spurious detections. The authors report a core detection accuracy of 96.6% on FVC2004 DB1.

Finally, in the approach by Zacharias et al. (2017), core detection starts with generating chain code contours from the thinned input fingerprint image. Chain-coded contours of the fingerprint ridges are then smoothed to improve robustness against noise and their curvature is computed using a rotation-invariant curvature estimation method. The core point is then localized on the ridge that bends the most.

Reference Direction Computation

Absolute pre-alignment with respect to rotation can be straightforward in full-size and/or high-quality fingerprints but critical for low-quality and partial fingerprint images. Some authors proposed to use (i) the shape of the external fingerprint silhouette (if available), (ii) the direction of the core delta segment (if a delta exists), (iii) the average orientation in neighborhoods around the core (Fig. 3.35), or (iv) the directions of the singularities (Bazen & Gerez, 2002). A specific geometric technique was proposed by Li et al. (2014) to measure the fingerprint direction of arch fingerprints using a set of isosceles triangles. Hotz (2009) showed that the vertical symmetry axis can be easily determined in arch fingerprints while the disturbance induced by the singularities can negatively affect the computation in non-arch fingerprints; therefore, he proposed a model-based cancellation of singularities to obtain an arch-like pattern from fingerprints of all the classes.

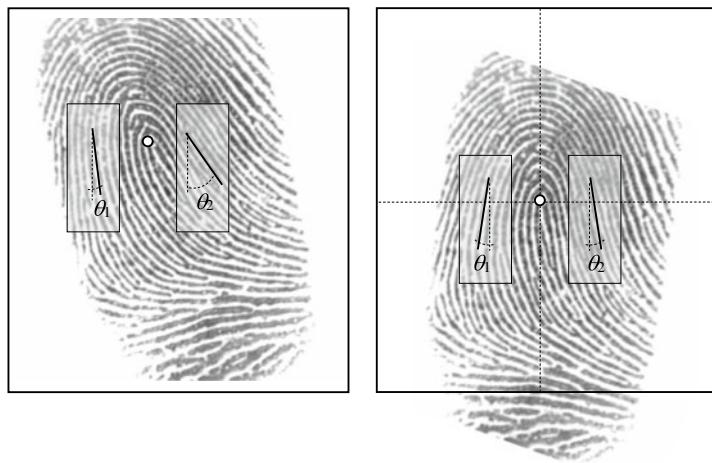


Fig. 3.35 The M82 method, developed for minutiae-based fingerprint matching in the AFIS community (Wegstein, 1982), performs a pre-alignment according to the core position and the average orientation of the two regions on the two sides of the core. The fingerprint on the right has been translated to move the core point to the image center and rotated to minimize the difference between the angles θ_1 and θ_2

Su et al. (2016) proposed to address fingerprint pose estimation as an object-detection problem. The finger detection method, which is based on the histogram of ridge orientation and support vector classifier, was used to compute both the center point and the direction of the fingerprint at the same time. This approach is shown to be effective on rolled fingerprints, but its application to plain (often partial) fingerprints remains challenging. Ouyang et al. (2017) proposed to use Faster R-CNN deep network model (Girshick, 2015) for fingerprint pose estimation and reported a better performance than Su et al. (2016). Another deep learning-based technique was proposed by Schuch et al. (2018a, b) where a Siamese CNN is trained to estimate the rotation between pairs of the sample. During training, a fingerprint is rotated by two random angles θ_1 and θ_2 . The CNN estimates rotations θ'_1 and θ'_2 , so that differences $(\theta_1 - \theta_2)$ and $(\theta'_1 - \theta'_2)$ are as similar as possible. While training takes place on fingerprints pairs, at inference time, the model receives a single input and provides its absolute direction.

Unlike singularities, fingerprint pose lacks a consistent definition, which makes it difficult to compare the methods of different research groups. A possible approach is to evaluate the pose consistency of different images of the same fingerprint (Su et al. 2016). Such evaluation is closely related to the application of pose in fingerprint indexing.

3.6 Enhancement

The performance of subsequent minutiae extraction algorithms and other fingerprint recognition techniques relies heavily on the quality of the input fingerprint images. In an ideal fingerprint image, ridges and valleys alternate and flow in a locally constant direction. In such situations, the ridges can be easily detected and minutiae can be precisely located in the image. Figure 3.36a shows an example of a good-quality fingerprint image. However, in practice, due to skin conditions (e.g., wet or dry, cuts, and bruises), sensor noise, finger pressure, and inherently low-quality fingertips (e.g., elderly people and manual workers), a significant percentage of fingerprint images (approximately 10%, according to our experience) is of poor quality like those in Figs. 3.36b, c. In many cases, a single fingerprint image contains regions of good, medium, and poor quality where the ridge pattern is very noisy and corrupted (Fig. 3.37). In general, there are several types of degradation associated with fingerprint images:

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

These three types of degradation make ridge extraction extremely difficult in the highly corrupted regions. This leads to the following problems in minutiae extraction: (i) a significant number of spurious minutiae are extracted, (ii) a large number of genuine minutiae are missed, and (iii) large errors in the location (position and orientation) of minutiae are introduced. In order to ensure good performance of the ridge and minutiae extraction

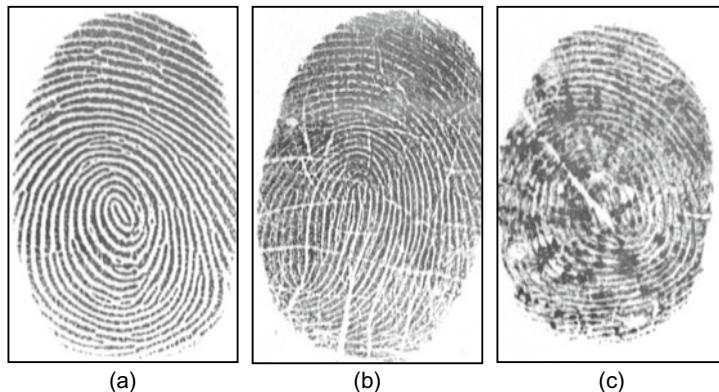
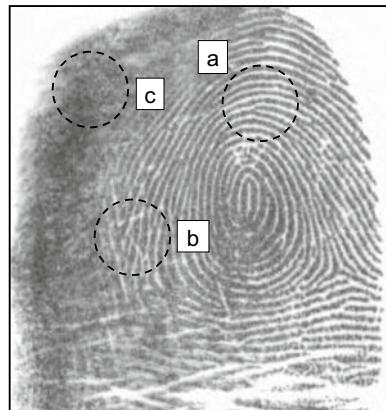


Fig. 3.36 **a** A good-quality fingerprint; **b** a medium-quality fingerprint characterized by scratches and ridge breaks; and **c** a poor-quality fingerprint containing a lot of noise

Fig. 3.37 A fingerprint image containing regions of different quality: **a** a well-defined region, **b** a recoverable region, and **c** an unrecoverable region



algorithms in poor-quality fingerprint images, an enhancement algorithm to improve the clarity of the ridge structure is necessary.

A human fingerprint expert is often able to correctly identify the minutiae by using various visual clues such as local ridge orientation, ridge continuity, ridge tendency, and so on. In theory, it is possible to develop an enhancement algorithm that exploits these visual clues to improve image quality. Generally, for a given fingerprint image, the fingerprint areas resulting from the segmentation step may be divided into three categories (Fig. 3.37):

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

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

A particular case of enhancement is super-resolution which consists in increasing the native resolution of a fingerprint image. If a fingerprint is captured at low resolution

(e.g., with a low-cost mobile sensor) with a resolution of 250 dpi, then its ridges are difficult to separate and minutiae detection is prone to errors. While a simple upscaling by interpolation leads to a blurred representation, effective super-resolution approaches have been proposed based on sparse representation by patch dictionaries (Singh et al., 2015 and Bian et al., 2017b).

3.6.1 Pixel-Wise Enhancement

In a pixel-wise operation, the new value of each pixel only depends on the previous value of that pixel and some global parameters (but not on the value of the neighboring pixels). Pixel-wise techniques do not produce satisfying results for fingerprint image enhancement. However, contrast stretching, histogram manipulation, normalization (Hong et al., 1998), and Wiener filtering (Greenberg et al., 2000) have been shown to be effective as initial processing steps in a more sophisticated fingerprint enhancement algorithm.

The normalization approach used by Hong et al. (1998) determines the new intensity value of each pixel in an image as

$$\mathbf{I}'[x, y] = \begin{cases} m_0 + \sqrt{(\mathbf{I}[x, y] - m)^2 \cdot v_0/v} & \text{if } \mathbf{I}[x, y] > m \\ m_0 - \sqrt{(\mathbf{I}[x, y] - m)^2 \cdot v_0/v} & \text{otherwise,} \end{cases} \quad (3.9)$$

where m and v are the image mean and variance, respectively, and m_0 and v_0 are the corresponding mean and variance after the normalization. Figure 3.38 shows an example. Since the mean and variance can change in different regions of a fingerprint image, the above global technique can be implemented in a local fashion: Kim and Park (2002) introduced a block-wise implementation of Eq. (3.9) where m and v are the block mean



Fig. 3.38 An example of normalization with the method described in Hong et al. (1998) using ($m_0 = 100$, $v_0 = 100$). © IEEE. Reprinted, with permission, from Hong et al. (1998)

and variance, respectively, and m_0 and v_0 are adjusted for each block according to the block features. A similar adaptive normalization was proposed by Shi and Govindaraju (2006b). However, this kind of normalization involves pixel-wise operations and does not change the ridge and valley structures. In particular, it is not able to fill small ridge breaks, fill intra-ridge holes, or separate parallel touching ridges.

A contrast enhancement scheme was proposed by Hari et al. (2013) where a high-pass filtered and scaled version of an image is added with itself. A quadratic filter, optimized for fingerprint images, is shown to be more effective than Laplacian and Laplacian of Gaussian filters.

Sharma and Dey (2019) proposed to adapt the preprocessing stage to the fingerprint image quality. For this purpose, fingerprints are clustered into five classes {dry, wet, normal dry, normal wet, or good} and the parameters of the preprocessing algorithm (in particular, the image sharpening) are tuned accordingly.

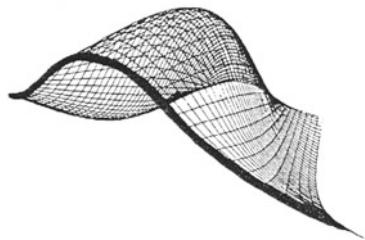
3.6.2 Contextual Filtering

The most widely used technique for fingerprint image enhancement is based on *contextual filters*. In conventional image filtering, only a single filter is used for convolution throughout the image. In contextual filtering, the filter characteristics change according to the local context. Usually, a set of filters is pre-computed and one of them is selected for each image region. In fingerprint enhancement, the context is often defined by the local ridge orientation and local ridge frequency. In fact, the sinusoidal-shaped wave of ridges and valleys is mainly defined by a local orientation and frequency that varies slowly across the fingerprint area. An appropriate filter that is tuned to the local ridge frequency and orientation can efficiently remove the undesired noise and preserve the true ridge and valley structure.

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

The output of a contextual fingerprint enhancement can be a gray-scale, near-binary, or binary image, depending on the filter parameters chosen. When selecting the appropriate set of filters and tuning their parameters, one should keep in mind that the goal is not to produce a good visual appearance of the image but to facilitate the robustness of the successive feature extraction steps. Since in poor-quality regions the estimation of the local context (orientation and frequency) may be erroneous, a too aggressive filtering is likely to produce spurious structures (Jiang, 2001). For example, an iterative application

Fig. 3.39 The shape of the filter proposed by O’Gorman and Nickerson (1989). © Elsevier. Reprinted, with permission, from O’Gorman and Nickerson (1989)



of the Gabor filters has been used by Cappelli et al. (2000) (refer to Chap. 7) to generate a synthetic fingerprint pattern; in this case, the filters generate nonexistent ridge patterns.

Contextual Filtering in the Spatial Domain

The method proposed by O’Gorman and Nickerson (1988, 1989) was one of the first to use contextual filtering for fingerprint enhancement; the authors defined a “mother filter” based on four main parameters of fingerprint images at a given resolution: minimum and maximum ridge width, and minimum and maximum valley width. The filter is bell-shaped (see Fig. 3.39), elongated along the ridge direction, and cosine tapered in the direction normal to the ridges. The local ridge frequency is assumed constant, and therefore, the context is defined only by the local ridge orientation. Once the mother filter has been generated, a set of 16 rotated versions (in steps of 22.5°) is derived. The image enhancement is performed by convolving each point of the image with the filter in the set whose orientation best matches the local ridge orientation. Depending on some input parameters, the output image may be gray-scale or binary. Examples of image binarization using this technique are shown in Figs. 3.49b and 3.50a.

Hong et al. (1998) proposed an effective method based on the Gabor filters, which today is still one of the most popular techniques for fingerprint enhancement. The Gabor filters have both frequency-selective and orientation-selective properties and have the optimal joint resolution in both spatial and frequency domains (Daugman, 1985; Jain & Farrokhnia, 1991). As shown in Fig. 3.40, a Gabor filter is defined by a sinusoidal plane wave (the second term of Eq. (3.10)) tapered by a Gaussian (the first term in Eq. (3.10)). The even symmetric two-dimensional Gabor filter has the following form.

$$g(x, y; \theta, f) = \exp\left\{-\frac{1}{2}\left[\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right]\right\} \cdot \cos(2\pi f \cdot x_\theta), \quad (3.10)$$

where θ is the orientation of the filter, and $[x_\theta, y_\theta]$ are the coordinates of $[x, y]$ after a clockwise rotation of the Cartesian axes by an angle of $(90^\circ - \theta)$.

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

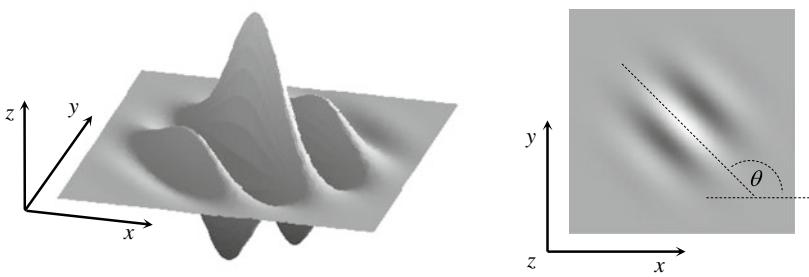


Fig. 3.40 Graphical representation (lateral view and top view) of the Gabor filter defined by the parameters $\theta = 135^\circ$, $f = 1/5$, and $\sigma_x = \sigma_y = 3$

In the above expressions, f is the frequency of a sinusoidal plane wave, and σ_x and σ_y are the standard deviations of the Gaussian envelope along the x - and y -axes, respectively.

To apply the Gabor filters to an image, the four parameters (θ , f , σ_x , and σ_y) must be specified. Obviously, the frequency of the filter is completely determined by the local ridge frequency and the orientation is determined by the local ridge orientation. The selection of the values σ_x and σ_y involves a trade-off. The larger the values, the more robust the filters are to the noise in the fingerprint image, but they are also more likely to create spurious ridges and valleys. On the other hand, the smaller the values, the less likely the filters are to introduce spurious ridges and valleys but then they will be less effective in removing the noise. In fact, from the Modulation Transfer Function (MTF) of the Gabor filter, it can be shown that increasing σ_x and σ_y decreases the bandwidth of the filter and vice versa. Based on empirical data, Hong et al. (1998) set $\sigma_x = \sigma_y = 4$. To make the enhancement faster, instead of computing the best-suited contextual filter for each pixel “on the fly,” a set $\{g_{ij}(x, y) | i = 1, \dots, n_o, j = 1, \dots, n_f\}$ of filters is a priori created and stored, where n_o is the number of discrete orientations $\{\theta_i | i = 1, \dots, n_o\}$ and n_f denotes the number of discrete frequencies $\{f_j | j = 1, \dots, n_f\}$. Then each pixel $[x, y]$ of the image is convolved, in the spatial domain, with the filter $g_{ij}(x, y)$ such that θ_i is the discretized orientation closest to θ_{xy} and f_j is the discretized frequency closest to f_{xy} . Figure 3.41 shows an example of the filter set for $n_o = 8$ and $n_f = 3$. Figure 3.42 shows the application of the Gabor-based contextual filtering on medium- and poor-quality images.

Greenberg et al. (2000) noted that by reducing the value of σ_x with respect to σ_y , the filtering creates fewer spurious ridges and is more robust to noise. In practice, reducing σ_x results in increasing the frequency bandwidth, independently of the angular bandwidth which remains unchanged; this allows the filter to better tolerate errors in local frequency estimates. Analogously, one could decrease σ_y in order to increase the angular bandwidth as pointed out by Sherlock et al. (1994). Their method increases the angular bandwidth near the singularities where the ridges are characterized by higher curvatures and the orientation changes rapidly. The methods by Erol et al. (1999); Wu and Govindaraju (2006); Van and Le (2009b) relate the filter bandwidth to the local orientation coherence,

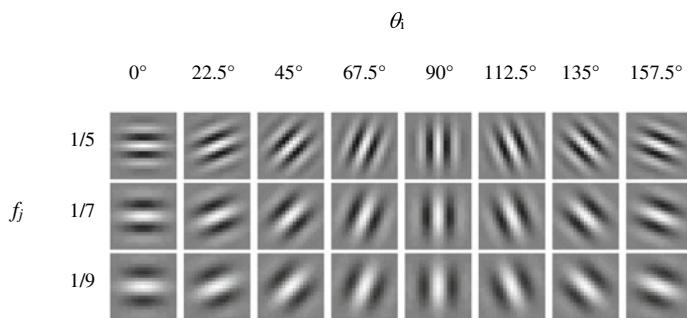


Fig. 3.41 A graphical representation of a bank of 24 ($n_o = 8$ and $n_f = 3$) Gabor filters where $\sigma_x = \sigma_y = 4$

whereas the Bernard et al. (2002) approach reduces the filter bandwidth if none of the responses to an initial set of filters exceeds a certain threshold. Yang et al. (2003) argue that the fingerprint ridge and valley pattern does not always resemble a pure sinusoidal pattern, mainly because of the different values of ridge and valley width in some regions (see Fig. 3.43). Therefore, they propose the Gabor-like filters whose positive and negative peaks can have different periods and contextually adjust the two periods based on the local ridge width and local valley width, respectively. Zhu et al. (2004) note that implementing the Gabor-based contextual filtering with a squared mask can lead to artifacts that can be removed if the mask support is circular.

Wang et al. (2008) suggest replacing the standard Gabor filter with the Log-Gabor filter to overcome the drawbacks that the maximum bandwidth of a Gabor filter is limited to approximately one octave and the Gabor filters are not optimal if one is seeking broad spectral information with maximal spatial localization. Curved Gabor filters have been introduced by Gottschlich (2012) to better enhance the fingerprint pattern in high curvature regions.

For low-cost and computation-limited fingerprint systems (e.g., embedded systems), the two-dimensional convolution of an image with a Gabor filter pre-computed over a discrete mask (e.g., 15×15) can be too time-consuming. The computational complexity can be reduced by using separable Gabor filters (Areekul et al., 2005) or masks with sparse coefficients (Jang et al., 2006).

Contextual Filtering in the Frequency Domain

Sherlock et al. (1992, 1994) performed contextual filtering in the Fourier domain; in fact, it is well known that a convolution in the spatial domain corresponds to a point-by-point complex multiplication in the Fourier domain (Gonzales & Woods, 2007). The filter is defined in the frequency domain by the function:

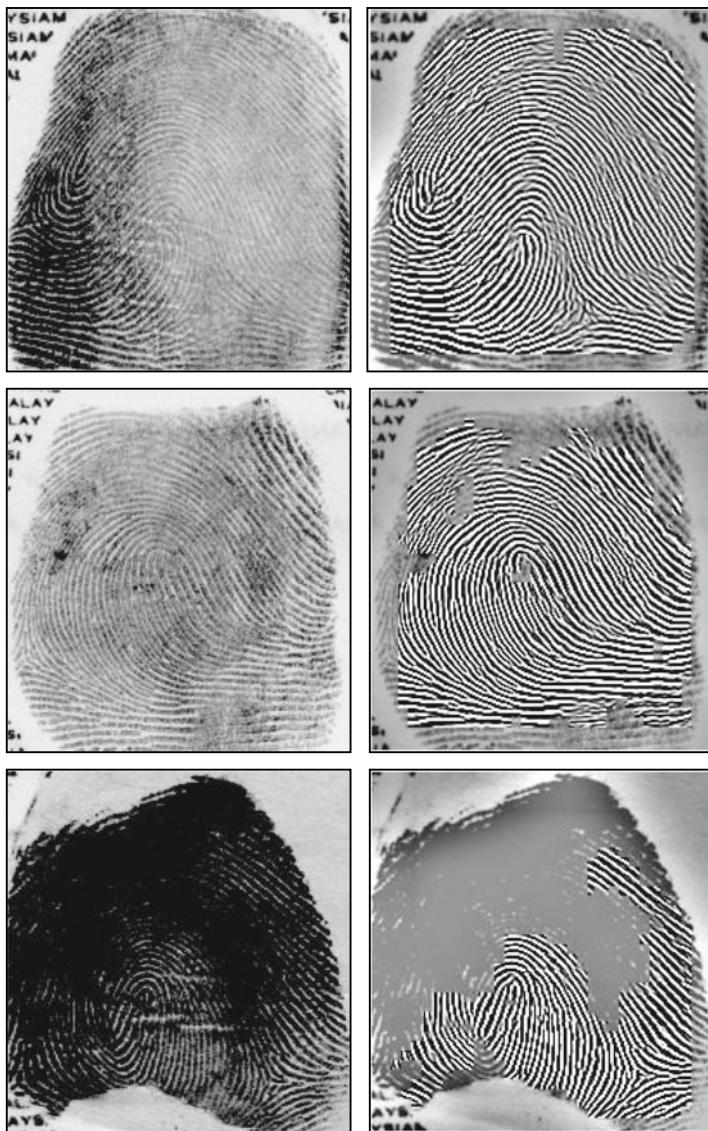


Fig. 3.42 Examples of fingerprint enhancement with the Gabor filtering as proposed by Hong et al. (1998). On the right, the enhanced recoverable regions are superimposed on the corresponding input images. © IEEE. Reprinted, with permission, from Hong et al. (1998)

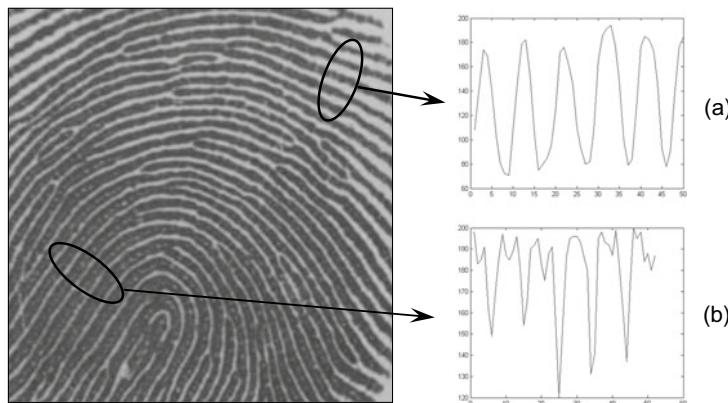


Fig. 3.43 Two examples of fingerprint regions where the local ridge–valley pattern conforms to **a** and deviates from **b** a sinusoidal pattern. © Elsevier. Reprinted, with permission, from Yang et al. (2003)

$$H(\rho, \theta) = H_{\text{radial}}(\rho) \cdot H_{\text{angle}}(\theta), \quad (3.11)$$

where H_{radial} depends only on the local ridge spacing $\rho = 1/f$ and H_{angle} depends only on the local ridge orientation θ . Both H_{radial} and H_{angle} are defined as bandpass filters and are characterized by a mean value and a bandwidth. A set of n discrete filters is derived by their analytical definition. To reduce the number of filters, only a single value is used for the local ridge frequency, and, therefore, the context is determined only by the orientation. The Fourier transform \mathbf{P}_i , $i = 1 \dots n$ of the filters is pre-computed and stored. Filtering an input fingerprint image \mathbf{I} is performed as follows (see Fig. 3.44).

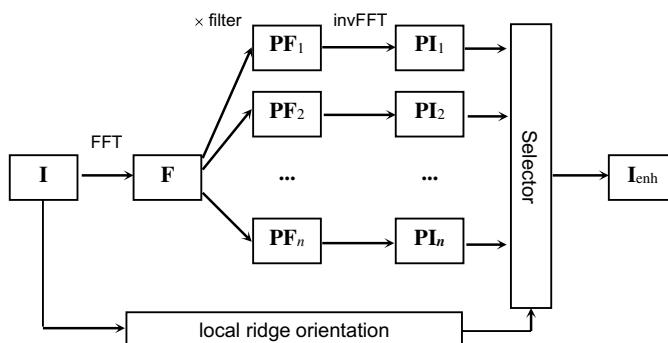


Fig. 3.44 Enhancement of the fingerprint image \mathbf{I} according to the Sherlock et al. (1994) method

- The FFT (Fast Fourier Transform) \mathbf{F} of \mathbf{I} is computed.
- Each filter \mathbf{P}_i is point-by-point multiplied by \mathbf{F} , thus obtaining n filtered image transforms \mathbf{PF}_i , $i = 1 \dots n$ (in the frequency domain).
- Inverse FFT is computed for each \mathbf{PF}_i resulting in n filtered images \mathbf{PI}_i , $i = 1 \dots n$ (in the spatial domain).

The enhanced image \mathbf{I}_{enh} is obtained by setting, for each pixel $[x,y]$, $\mathbf{I}_{enh}[x, y] = \mathbf{PI}_k[x, y]$, where k is the index of the filter whose orientation is the closest to θ_{xy} .

Chikkerur et al. (2007) proposed an efficient implementation of contextual filtering based on Short-Time Fourier transform (STFT) that requires partitioning the image into small overlapping blocks and performing the Fourier analysis separately on each block. The orientation and frequency of each block are probabilistically determined through Eqs. (3.6) and (3.8), and the orientation coherence is computed similar to Eq. (3.4). Each block is then filtered (by complex multiplication in the Fourier domain) with a filter equivalent to Eq. (3.11) except for the angular bandwidth which is adjusted according to the orientation coherence; in Sherlock et al. (1994), the angular bandwidth is related to the distance from the closest singular point. Since singular point estimation is less robust than coherence estimation, the Chikkerur et al. (2007) bandwidth adjustment seems to be more effective than the approach by Sherlock et al. (1994).

An approach similar to that of Chikkerur et al. (2007) was introduced by Jirachaweng and Areekul (2007), but their block-wise contextual information computation and filtering are performed in the DCT (Discrete Cosine Transform) domain instead of in the Fourier domain. Finally, in the method introduced by Bartunek et al. (2013), contextual filtering is performed in the spatial domain by using cosine functions aligned to the frequency and orientation of the dominant peaks extracted in the frequency domain.

Robust Filter Selection

The need for an effective enhancement is particularly important in poor-quality fingerprints where only the recoverable regions carry information necessary for matching. On the other hand, computing local information (context) with sufficient reliability in poor-quality fingerprint images is very challenging (see Fig. 3.45). To overcome this problem, Kamei and Mizoguchi (1995), Hong et al. (1996), Bernard et al. (2002), and Nakamura et al. (2004) proposed to apply all the filters of a given set at each point in the image (as in Fig. 3.44). A “selector” then chooses the best response from all the filter responses:

- In the method by Kamei and Mizoguchi (1995), the selection is performed by minimizing an energy function that includes terms that require orientation and frequency to be locally smooth.
- Hong et al. (1996) and Nakamura et al. (2004) base their selection on the analysis of local ridges extracted from the filtered images. In particular, Nakamura et al. (2004) enforce orientations that are consistent with a ridge parallelism model.

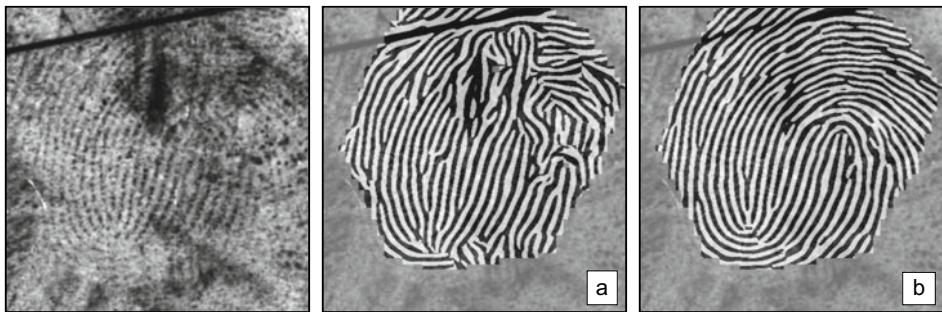


Fig. 3.45 Enhancement of a noisy latent fingerprint with Gabor filtering: **a** the contextual information (orientation and frequency) are automatically computed; **b** the contextual information are provided through manual markup. The much higher quality of the enhancement in **(b)** denotes the importance of using reliable contextual data. For details, refer to Cappelli et al. (2009)

- Bernard et al. (2002) make the selection according to the maximum response. However, unlike most of the Gabor-based methods, phase information coming from the real and the imaginary part of Gabor filters is also used for the final image enhancement.

As expected, approaches that require convolution of an image with a large number of filters are computationally expensive, and it is difficult to obtain efficient implementations. This problem can be partially alleviated by exploiting steerable filter (Freeman & Adelson, 1991) which filters the image with a reduced number of basis filters and derives the remaining filter responses by a linear combination.

Matched filtering is another interesting technique that is able to perform a sort of contextual filtering without explicitly computing local ridge orientation and frequency. In the context of fingerprint enhancement, it was first proposed by Watson et al. (1994) and Willis and Myers (2001). Each 32×32 image block is enhanced separately; the Fourier transform of the block is multiplied by its power spectrum raised to a power k :

$$\mathbf{I}_{enh}[x, y] = F^{-1} \left\{ F(\mathbf{I}[x, y]) \times |F(\mathbf{I}[x, y])|^k \right\}. \quad (3.12)$$

The power spectrum contains information about the underlying dominant ridge orientation and frequency and the multiplication has the effect of enhancing the block accordingly. Watson et al. (1994) set $k = 0.6$ whereas Willis and Myers (2001) proposed a more aggressive value of $k = 1.4$. Unfortunately, to avoid discontinuities at the edges between adjacent blocks, a large amount of overlap between the neighboring blocks (e.g., 24 pixels) is necessary and this significantly increases the enhancement time.

3.6.3 Multi-Resolution and Iterative Enhancement

Multi-resolution analysis has been proposed to remove noise from fingerprint images. Decomposing the image into different frequency bands (or sub-images) allows us to compensate for different noise components at different scales: in particular, at higher noise levels (low and intermediate frequency bands), the rough ridge–valley flow is cleaned and gaps are closed, whereas at the lower levels (higher frequencies), the finer details are preserved. The enhanced image bands are then recombined to obtain the final image.

- The Almansa and Lindeberg (2000) technique performs shape-adapted smoothing based on second-moment descriptors and automatic scale selection (over a number of scales) based on normalized derivatives. The smoothing operation is adapted according to the local ridge structures, allowing interrupted ridges to be joined. The scale selection procedure estimates local ridge width and adapts the amount of smoothing to the local noise.
- In Hsieh et al. (2003), the multi-resolution representation is based on wavelet decomposition (Mallat, 1989). Each sub-image is processed through both textural and directional filtering to suppress spectral noise and to close the gaps produced by creases and scars.
- The Cheng and Tian (2004) method is based on dyadic space scale and the decomposition depth is determined according to the average ridge width in the image; the noise reduction relies on smoothing the differential images between successive scales (i.e., the details that are lost passing from one scale to the successive one).
- Fronthaler (2007, 2008a) use a Laplacian-like image-scale pyramid to decompose the original fingerprint into three smaller images corresponding to different frequency bands. Each image is then processed through contextual filtering.

The idea behind iterative enhancement is to first filter the fingerprint regions where contextual information is reliable enough to avoid introducing artifacts. Such early filtering determines an enlargement of these reliable regions thus enabling to properly process their neighbors. The entire process can be seen as a wildfire expansion which terminates once the filtered regions collapse and cover the entire image.

- Cappelli et al. (2012) proposed contextual iterative filtering that selectively applies a Gabor filter-bank (six orientations and three frequencies), starting from high-quality regions and then iteratively expanding to low-quality regions (see Fig. 3.46). This approach does not require any prior contextual information: all responses from the Gabor filter-bank are calculated and the enhanced pixels with strong response and high ridge flow homogeneity are selected for the enhanced fingerprint. The algorithm continues iterating until the maximum number of iterations is reached or no more low-quality regions exist in the image.



Fig. 3.46 Iterative enhancement of a fingerprint image as proposed by Cappelli et al. (2012). © IEEE. Reprinted, with permission, from Cappelli et al. (2012)

- The algorithm proposed by Sutthiwichaiporn and Areekul (2013) applies a Gaussian-matched filter (see Eq. 3.12) starting from high-quality regions and then iteratively propagating good spectra of enhanced ridges to lower quality regions. Spectra diffusion is also a fundamental component of the methods by Bian et al. (2018) where the Gabor filtering and Linear Contrast Stretching are performed in advance to improve robustness.
- Orientation diffusion, formerly adopted by some researchers to smooth local orientations (see Sect. 3.3.4), was applied to fingerprint enhancement by Zhao et al. (2009) and Gottschlich and Schönlieb (2012).
- Yang et al. (2013) proposed a two-stage schema: contextual filtering in the spatial domain is performed in the first stage and bandpass filtering in the frequency domain takes place in the second stage; information available at the end of the first stage are exploited to better tune the second stage filters (e.g., improvement of the contextual information).

3.6.4 Learning-Based Enhancement

One of the first fingerprint enhancement methods with a learning stage was proposed by Rama and Namboodiri (2011): this technique makes use of previously learned prior patterns from a set of clean fingerprints to restore a noisy one. A hierarchical interconnected Markov Random Field is used to process the information at multiple resolutions.

A CNN-based approach was proposed by Li et al. (2018) and successfully applied to latent fingerprints. The model consists of an encoder-decoder architecture composed of one common convolution part (encoder) shared by two different deconvolution branches (decoders), which are the enhancement and the orientation branch (see Fig. 3.47). While the enhancement branch is aimed at removing structured noise and providing an enhanced fingerprint image as output, the orientation branch is exploited to guide the enhancement

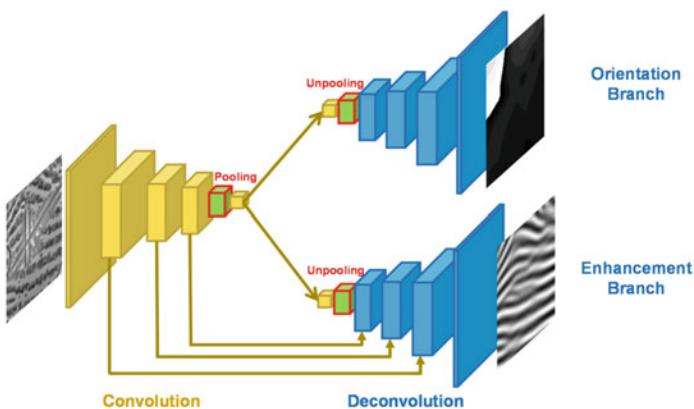


Fig. 3.47 The architecture proposed by Li et al. (2018). It is worth noting that the receptive field of a neuron after the last convolutional layer of the encoder (i.e., the fourth layer) is 53×53 pixels thus providing a large amount of contextual information. © Elsevier. Reprinted, with permission, from Li et al. (2018)

through a multi-task learning strategy. The network is trained end-to-end with a loss function working at the pixel level. The training examples are good-quality fingerprint patches corrupted by artificial structured noise; ground truth data for supervised training can be reliably obtained given the good quality of patches before alteration.

A CNN model similar to FingerNet was introduced by Wong and Lai (2020). Here too, a second parallel branch is used to reconstruct the local orientations and guide the enhancement process, but the reconstructed orientations are then fused with the pre-enhanced image for further improvement, resulting in a deeper architecture. Furthermore, training data is obtained by synthetic fingerprint generation, using the SFinGe method discussed in Chap. 7, which can output both the native good-quality fingerprints and the corresponding noise corrupted versions.

Latent fingerprint enhancement was modeled as an image-to-image translation problem (Isola et al., 2017) and solved with a conditional GAN approach by Joshi et al. (2019). The GAN model consists of two networks: an enhancer network and a discriminator network; the former has an encoder-decoder architecture and is trained to produce an enhanced version of the given noisy fingerprint image and the latter is trained to classify whether the input is a real enhanced image or a generated one. Since image-to-image translation requires paired training data (i.e., for each noisy latent fingerprint, it requires the corresponding high-quality ground truth), synthetic generation by SFinGe is used to prepare the training data.

Deep models based on autoencoders were also proposed for fingerprint enhancement by Schuch et al. (2016) and Svoboda et al. (2017).

3.6.5 Crease Detection and Removal

Some fingerprints (the incidence is higher in elderly people) are affected by the presence of a large number of creases (see Fig. 3.7c). The presence of creases adversely influences the computation of orientation image and can lead to the detection of false ridge-ending minutiae. Provided that contextual information (i.e., local orientation and frequency) has been correctly estimated, contextual filtering techniques are usually capable of filling the small ridge-gap produced by creases; however, if the image is very noisy or the creases are too wide, contextual filtering can fail. To explicitly detect (and optionally remove) creases, some ad hoc techniques have been proposed:

- Vernon (1993) argued that creases are characterized by collinear terminations on ridges and proposed a detection approach based on the analysis of the Hough transform space derived from the ridge-ending minutiae. The Hough transform (Ballard, 1981) is in fact a simple but powerful technique aimed to detect lines in noisy data.
- Wu et al. (2003) and Zhou et al. (2004) modeled a crease by using a parameterized rectangle, followed by a multi-channel filtering framework to detect creases at different orientations. Principal Component Analysis is applied to estimate the crease orientation, length, and width. Figure 3.48 shows some examples of crease detection.
- Oliveira and Leite (2008) identify crease points by looking at the discordance between local orientations computed at two different scales. In fact, when computed at fine scale (i.e., on a small neighborhood), the local orientation within a crease markedly deviates from the overall underlying ridge-line orientation that can be estimated at the coarse scale (i.e., on a large neighborhood). An approach based on Watershed transform is then proposed to remove the creases.
- Zhou et al. (2009b) designed an optimal crease detection filter shaped as a second-order Gaussian derivative. Besides using the detected creases to filter spurious minutiae in



Fig. 3.48 Two examples of crease detection (in black) by Wu et al. (2003) approach. © IEEE. Reprinted, with permission, from Wu et al. (2003)

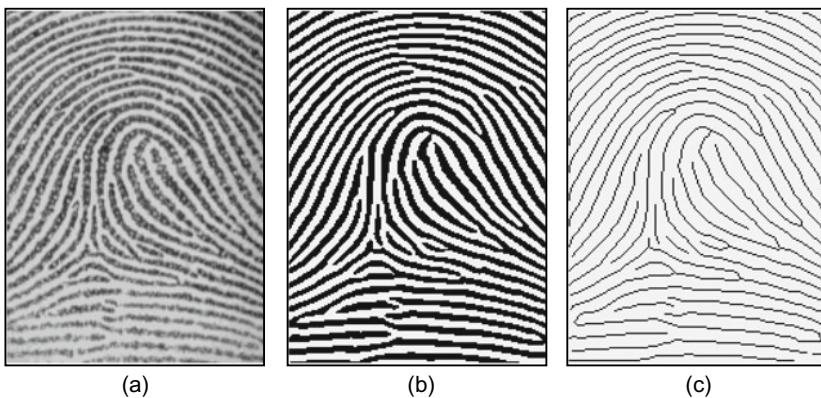


Fig. 3.49 **a** A fingerprint gray-scale image, **b** the image obtained after binarization of the image in **(a)**, and **c** skeleton image obtained after a thinning of the image in **(b)**. © IEEE. Reprinted, with permission, from Maio and Maltoni (1997)

their proximity, the authors showed that using the creases as features in combination with minutiae can improve fingerprint recognition in elderly people. A similar Gaussian derivative filter was used by Khan et al. (2016), but in this approach, the detected creases are then filled by an inpainting process guided by local orientations.

- Gottschlich et al. (2009) proposed a local-orientated detection approach that traces ridges and valleys on a binarized image. Groups of traced parallel lines are considered to determine local orientations. Traces can be also detected on creases but the number of parallel lines here is small compared to the genuine ridge and valley pattern, and therefore, creases can be quite easily discriminated.

3.7 Minutiae Detection

Most automated systems for fingerprint comparison are based on minutiae matching (see Chap. 4); hence, reliable minutiae extraction is an extremely important task and a substantial amount of research has been devoted to this topic. Most of the proposed methods require the fingerprint gray-scale image to be converted into a binary image. Some binarization processes greatly benefit from an a priori enhancement (see Sect. 3.6); on the other hand, some enhancement algorithms directly produce a binary output, and therefore, the distinction between enhancement and binarization sometimes vanishes. The binary images are usually submitted to a thinning stage which reduces the ridge-line thickness to one pixel, resulting in a skeleton image (Fig. 3.49). A simple image scan then allows the detection of pixels that correspond to minutiae.

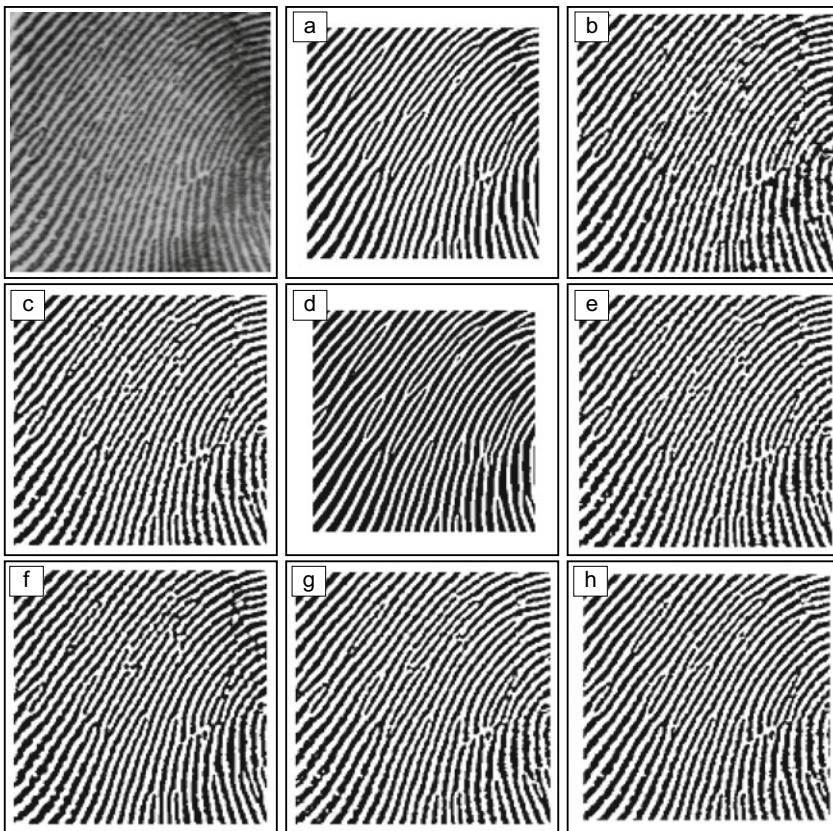


Fig. 3.50 A portion of a good-quality fingerprint image and its binarization through some of the early ad hoc binarization methods **a** O’Gorman and Nickerson (1989); **b** Verma et al. (1987); **c** local threshold approach; **d** Sherlock et al. (1994); **e** Xiao and Raafat (1991); **f** Moayer and Fu (1986); **g** Stock and Swonger (1969); and **h** Watson et al. (1994)

Some authors proposed minutiae extraction approaches that work directly on the gray-scale images without binarization and thinning. This choice was motivated by the following considerations:

- A significant amount of information may be lost during the binarization process and thinning may introduce a large number of spurious minutiae.
- Binarization and thinning can be time-consuming. This is not a serious concern with today’s fast processors but things were different some decades ago.
- In the absence of an a priori enhancement step, most of the binarization techniques do not provide satisfactory results when applied to low-quality images.

Recently, learning-based techniques have been proposed where minutiae detection is approached holistically, that is all the prior steps required by a classical pipeline (computation of local orientation and frequency, enhancement, binarization, thinning, etc.) are performed by a single model trained end-to-end.

3.7.1 Binarization-Based Methods

The general problem of image binarization has been widely studied in the fields of image processing and pattern recognition (Trier & Jain, 1995). The simplest approach uses a *global threshold* t and works by setting the pixels whose gray level is lower than t to 0 and the remaining pixels to 1.

In general, different portions of an image may be characterized by a different contrast and intensity and, consequently, a single threshold for the entire image is not sufficient for a correct binarization. For this reason, the *local threshold* technique changes t locally, by adapting its value to the average local intensity. In the specific case of fingerprint images, which are sometimes of very poor quality, a local threshold method cannot always guarantee acceptable results and more effective fingerprint-specific solutions are necessary. Therefore, a number of ad hoc binarization approaches have been introduced in the period 1969–1999. Since most of the modern fingerprint enhancement techniques (discussed in Sect. 3.6.2) produce a clear binary image for appropriately chosen parameters,⁴ the former ad hoc binarization techniques are today of scarce practical relevance. Figure 3.50 shows a qualitative comparison of binarization results obtained by some of the early methods.

Minutiae detection from binary images is usually performed after an intermediate thinning step that reduces the width of the ridges to one pixel. Thinning algorithm is critical and the aberrations and irregularity of the binary-ridge boundaries have an adverse effect on the *skeletons* (i.e., the one-pixel-width ridge structure), resulting in “hairy” growths (spikes) that lead to the detection of spurious minutiae. With the aim of improving the quality of the binary images before the thinning step, some researchers have introduced regularization techniques which usually work by filling holes (see Fig. 3.51), removing small breaks, eliminating bridges between ridges, and other artifacts. For this purpose, Coetzee and Botha (1993) identify holes and gaps by tracking the ridge-line edges through adaptive windows and removing them using a simple blob-coloring algorithm. Hung (1993) uses an adaptive filtering technique to equalize the width of the ridges; narrow ridges in under-saturated regions are expanded and thick ridges in over-saturated regions are shrunk. Wahab et al. (1998) correct the binary image at locations where orientation estimates deviate from their neighboring estimates. This correction is performed by substituting the noisy pixels according to a set of oriented templates. Luo and Tian (2000) implement a two-step method, where the skeleton extracted at the end of the first step is

⁴ In any case, even when the output of the enhancement is a gray-scale image, a simple local thresholding technique often results in satisfactory binarization.

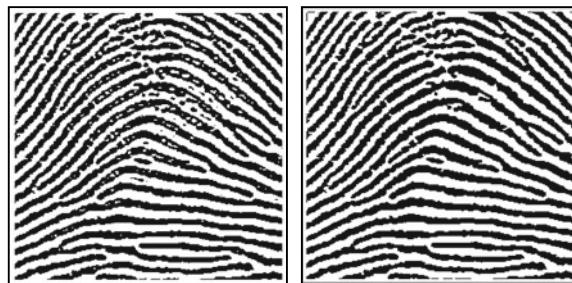


Fig. 3.51 The result of eliminating small holes from both the ridges and valleys of the binary image; the input image is shown on the left and the output is shown on the right. The filtering is performed by computing the connected components of the image and by removing the components whose area (number of pixels) is smaller than a given threshold

used to improve the quality of the binary image based on a set of structural rules; a new skeleton is then extracted from the improved binary image. Finally, in the method introduced by Bhowmick and Bhattacharya (2009), the binary skeleton is regularized through cubic B-splines fitting.

Mathematical morphology (Gonzales & Woods, 2007) is a powerful and elegant tool of digital topology that allows a regularization of the shape of binary objects. Some authors propose morphology-based techniques for regularizing binary fingerprint images:

- Fitz and Green (1996) and Ikeda et al. (2002) remove small lines and dots both in the ridges and valleys of binary images through the application of morphological operators.
- To remove the spikes that often characterize the thinned binary images, Ratha et al. (1995) implement a morphological “open” operator whose structuring element is a small box oriented according to the local ridge orientation.
- Liang and Asano (2006) recommend using Generalized Morphology Operators (GMO) that may increase the robustness of the algorithms to noise and small intrusions, especially when medium-size structuring elements are used. An efficient implementation of GMO-based techniques for removing salt and pepper noise and small islands is proposed based on the distance transform (Gonzales & Woods, 2007) and the integral image (Viola & Jones, 2001).

As far as thinning techniques are concerned (Lam et al., 1992), a large number of approaches are available in the literature due to the central role of this processing step in many pattern recognition applications: character recognition, document analysis, and map and drawing vectorization. Hung (1993) used the algorithm by Arcelli and Baja (1984); Ratha et al. (1995) adopted a technique included in the HIPS library (Landy et al., 1984); Mehtre (1993) employed the parallel algorithm described in Tamura (1978); and Coetzee and Botha (1993) used the method by Baruch (1988). In Ji et al. (2007), the skeleton

is computed through a constrained PCNN (Pulse Coupled Neural Network) where the orientation image is used to constrain the thinning direction of PCNN thus allowing to reduce bothersome artifacts such as the short spikes that conventional thinning algorithms often produce. Sudiro et al. (2007) noted that in the binarized image, valleys are often thinner than ridges, and since the time taken by a thinning algorithm increases with the initial thickness of the objects, they propose extracting minutiae from valleys to reduce the computation time. Further fingerprint-specific thinning algorithms have been proposed by Xiang et al. (2009); Wang et al. (2011); and Saleh et al. (2009).

Once a binary skeleton has been obtained, a simple image scan allows the pixels corresponding to minutiae to be detected according to the ANSI/NIST-ITL 1-2011 (2015) coordinate models shown in Fig. 3.5; in fact, the pixels corresponding to minutiae are characterized by a *crossing number* different from 2. The crossing number $cn(\mathbf{p})$ of a pixel \mathbf{p} in a binary image is defined (Arcelli & Baja, 1984) as half the sum of the differences between pairs of adjacent pixels in the eight-neighborhood of \mathbf{p} :

$$cn(\mathbf{p}) = \frac{1}{2} \sum_{i=1 \dots 8} |val(\mathbf{p}_{i \text{ mod } 8}) - val(\mathbf{p}_{i-1})|,$$

where $\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_7$ are the pixels belonging to an ordered sequence of pixels defining the eight-neighborhood of \mathbf{p} and $val(\mathbf{p}) \in \{0,1\}$ is the pixel value. It is simple to note (Fig. 3.52) that a pixel \mathbf{p} with $val(\mathbf{p}) = 1$ has the following properties:

- Is an intermediate ridge point if $cn(\mathbf{p}) = 2$.
- Corresponds to a ridge-ending minutia if $cn(\mathbf{p}) = 1$.
- Corresponds to a bifurcation minutia if $cn(\mathbf{p}) = 3$.
- Defines a more complex minutia (e.g., crossover) if $cn(\mathbf{p}) > 3$.

Some techniques have been proposed in the literature to extract minutiae from binary images without using the crossing number to check the pixel connectivity on the skeleton resulting from a thinning step: Leung et al. (1991) method extracts the minutiae from

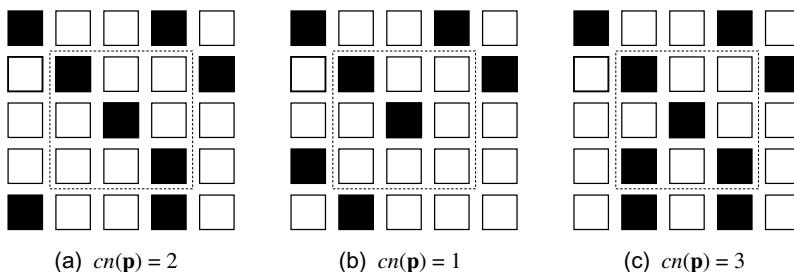


Fig. 3.52 **a** Intra-ridge pixel; **b** ridge-ending minutia; **c** bifurcation minutia

thinned binary images using a three-layer perceptron neural network. The algorithm by Gamassi et al. (2005) is a variant of the crossing number method that can work with thick binary ridges; in fact, for each point, the algorithm counts the black–white transitions along a square path centered at that point and large enough to touch two ridges. Approaches by Weber (1992); Govindaraju et al. (2003); and Shi and Govindaraju (2006a) work on thick binary ridges and exploit special ridge tracking algorithms. Shin et al. (2006) encode thick binary ridges with Run Length Code (RLC) and extract minutiae by searching for the termination or bifurcation points of ridges in the RLC. Miao et al. (2007); Zhang et al. (2011b); and Ma and Zhu (2013) encode the skeleton by means of principal curves which are self-consistent smooth curves suitable to approximate noisy data. Finally, the foundation of the method developed by Székely and Székely (1993) lies in the use of a divergence operator capable of discerning fingerprint pattern discontinuities that correspond to minutiae.

Minutiae angle or direction θ (see Fig. 3.5), in addition to the minutiae coordinates, is used by most of the matching algorithms, to enforce minutiae pairing or correspondence between two fingerprint images. A simple way to determine the minutiae direction is to start from the local ridge orientation at the minutia origin and to convert this orientation into a direction (i.e., deciding the quadrant) by looking at the departing ridge(s).

3.7.2 Direct Gray-Scale Extraction

With the aim of overcoming some of the problems related to fingerprint image binarization and thinning (e.g., the presence of spurious minutiae in the case of irregular ridge edges), some authors have proposed direct gray-scale extraction methods.

Detection Based on Ridge Tracing

Maio and Maltoni (1997) proposed a direct gray-scale minutiae extraction technique, whose basic idea is to track the ridge lines in the gray-scale image, by “sailing” according to the local orientation of the ridge pattern. From a mathematical point of view, a ridge line is defined as a set of points that are local maxima along one direction. The ridge-line extraction algorithm attempts to locate, at each step, a local maximum relative to a section orthogonal to the ridge direction. By connecting the consecutive maxima, a polygonal approximation of the ridge line can be obtained. Given a starting point $[x_c, y_c]$ and a starting direction θ_c , the *ridge line following* algorithm (see Fig. 3.53) computes a new point $[x_t, y_t]$ at each step by moving μ pixels from the current point $[x_c, y_c]$ along the direction θ_c . Then it computes the *section set* Ω as the set of points belonging to the section segment lying on the xy -plane with a median point $[x_t, y_t]$, direction orthogonal to θ_c , and length $2\sigma + 1$. A new point $[x_n, y_n]$, belonging to the ridge line, is chosen among the local maxima of an enhanced version of the set Ω . The point $[x_n, y_n]$ becomes the current point $[x_c, y_c]$ and a new direction θ_c is computed (Fig. 3.53). The optimal

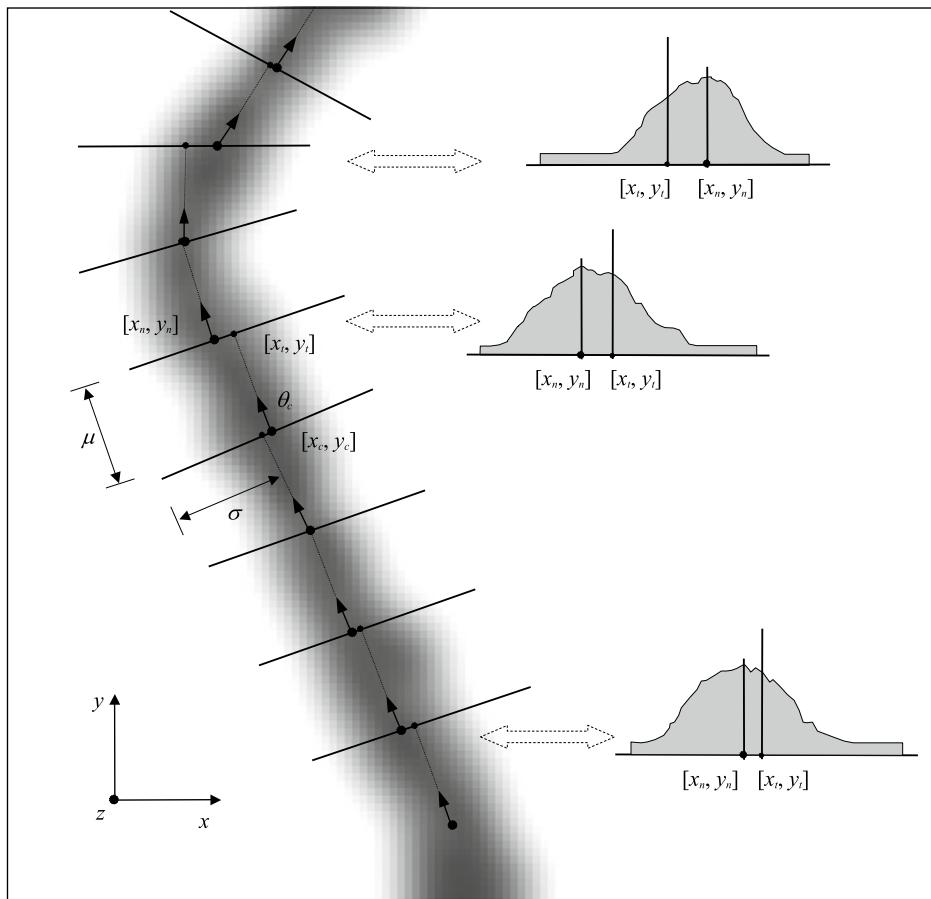


Fig. 3.53 Some ridge line following steps (Maio & Maltoni, 1997). On the right, some sections of the ridge line are shown. © IEEE. Reprinted, with permission, from Maio and Maltoni (1997)

value of the parameters μ and σ can be determined according to the average thickness of the ridge lines. Figure 3.54 shows the results obtained by applying the minutiae detection algorithm to a sample fingerprint.

Jiang et al. (1999, 2001) proposed a variant of the Maio and Maltoni (1997) method, where the ridge following step μ is dynamically adapted to the change of ridge contrast and bending level. Referring to Fig. 3.53, a large step value is selected if there is little variation in the gray-level intensity along the segment $[x_c, y_c]$ $[x_t, y_t]$, and the ridge bending is low. On the other hand, a high bending level of the ridge (possibly facing a ridge bifurcation) or large intensity variations (possibly facing a ridge ending) will result in a small step value. Using a dynamic step speeds up the tracing while maintaining good precision.

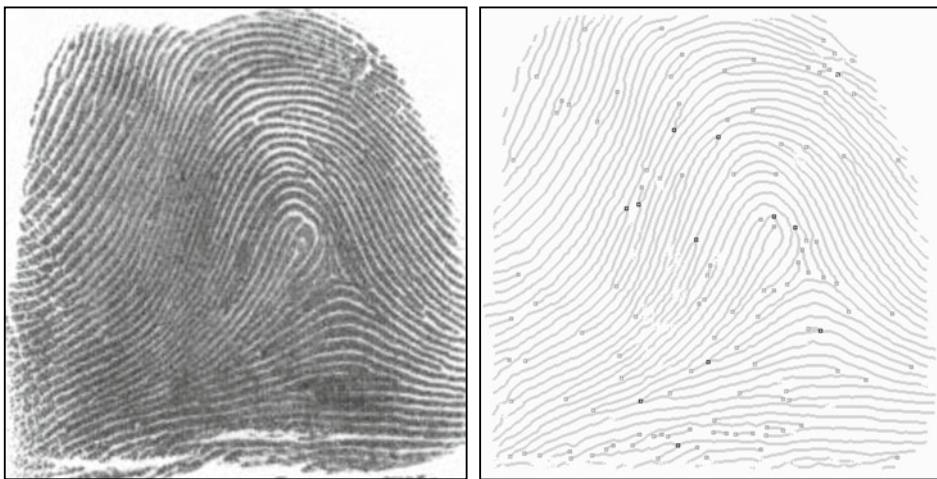


Fig. 3.54 Minutiae detection on a sample fingerprint by using the Maio and Maltoni (1997) method. Ridge-ending minutiae are denoted by gray boxes and bifurcation minutiae are denoted by black boxes

Liu et al. (2000) introduced another modification of the Maio and Maltoni (1997) method. Instead of tracking a single ridge, the algorithm simultaneously tracks a central ridge and the two surrounding valleys. For this purpose, they search a central maximum and two adjacent minima in each section Ω . Minutiae are detected when the shape of the section deviates from a configuration with a central maximum and two side minima. Here too, the ridge following step μ is dynamically adjusted according to the distances between lateral minima from the central maximum. This approach does not need an a priori setting of some parameters such as the maximum bending angle (which determines a stopping criterion in the original algorithm) and the step value μ .

The Chang and Fan (2001) approach is aimed at discriminating the true ridge maxima in the sections Ω obtained during ridge line following. Two thresholds are initially determined based on the gray-level histogram decomposition. The histogram is modeled as a sum of three Gaussian contributions associated with the background, valleys, and ridges, respectively. The mean, variance, and probability of each Gaussian are estimated and two thresholds are derived for successive characterization of maxima and minima of the section Ω . A set of rules is employed to discriminate real ridge points from background noise and intra-ridge variations.

Finally, an efficient integer-arithmetic version of Maio and Maltoni (1997) algorithm has been developed by Canyellas et al. (2005) whose main aim was to port it to low-cost hardware.

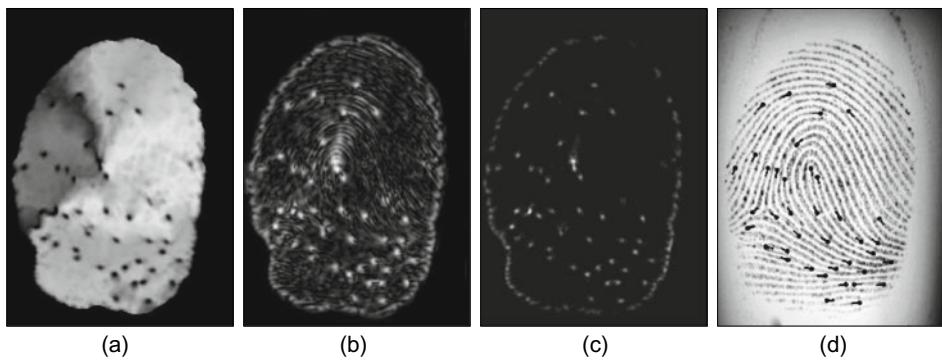


Fig. 3.55 Application of the minutiae detection method proposed by Fronthaler et al. (2008a) to the fingerprint image (d). **a** Linear symmetry (LS), **b** parabolic symmetry (PS), **c** PS ($1 - |LS|$), and **d** minutiae detected as local maxima of (c) superimposed to the original fingerprint image. Images courtesy of J. Bigun

Detection Based on Discontinuities of the Ridge–Valley Flow

Nilsson and Bigun (2001) proposed using Linear Symmetry (LS) properties computed by spatial filtering (Bigun & Granlund, 1987) via separable Gaussian filters and Gaussian derivative filters. Minutiae are identified in the gray-scale image as points characterized by the lack of symmetry. In fact, whereas a non-minutia point is characterized by a direction (i.e., the minutiae angle) along which an infinitesimal translation leaves the pattern least variant, minutiae are local discontinuities in the LS vector field. Fronthaler et al. (2008a) used both Linear Symmetry (LS) and Parabolic Symmetry (PS) to detect minutiae. Near a minutia point, the response to a parabolic symmetry filter is high while the response to a linear symmetry filter is low; hence, the expression PS ($1 - |LS|$) allows to detect minutiae more reliably than PS or LS alone. Figure 3.55 shows an example.

Liu and Cao (2016) demonstrated that minutiae can be extracted from Level-1 features, in particular from local orientations and frequencies computed with sufficient resolution. Their approach is based on the FM (Frequency Modulation) model introduced by Larkin and Fletcher (2007) which is discussed in more detail in Chap. 7. In the FM model, a fingerprint image can be represented as a function of a composite phase which is the sum of two components: a continuous phase which is smooth and changes slightly and a spiral phase containing singularities in correspondence of minutiae points. The computation of curve integrals along small circles⁵ allows to detect singularities in the composite phase and therefore to extract minutiae points. While this method is very interesting from a theoretical point of view, its accuracy and efficiency are lower than techniques extracting minutiae from ridges.

⁵ For analogy, see the Poincaré method in Sect. 3.5.1.

3.7.3 Learning-Based Approaches

Early minutiae detection approaches based on neural networks were proposed by Leung et al. (1990) and Wahab et al. (2004). In Leung et al. (1990), a multilayer perceptron analyzes the output of a rank of the Gabor filters applied to the gray-scale image. The image is first transformed into the frequency domain where the filtering takes place; the resultant magnitude and phase signals constitute the input to a neural network composed of six sub-networks, each of which is responsible for detecting minutiae at a specific orientation; a final classifier is employed to combine the intermediate responses.

One of the first techniques in the deep learning era was introduced by Sankaran et al. (2014) for latent fingerprints. Latent fingerprint minutiae extraction is posed as a binary classification problem to classify (64×64 pixels) patches as minutia or non-minutia. Minutia and non-minutia descriptors are learnt from a large number of tenprint fingerprint patches using stacked denoising sparse autoencoders. Jiang et al. (2016) also proposed a minutiae detection method working at the patch level. This method is based on two stages: in the former, a CNN-based classifier selects patches likely containing one or more minutiae in their central part; in the latter, a second CNN locates in the pre-selected patches the internal subwindows containing minutiae.

A further patch-based approach was introduced by Darlow and Rosman (2017). Here a single CNN is used for classification and some post-processing steps are implemented to improve the location accuracy. The most relevant contribution of this work is the introduction of an automatic procedure to establish minutiae ground truth, based on a “smooth” fusion of the responses of five commercial minutiae extractors.

The main limitations of patch-based approaches are (i) the low efficiency due to the need of a sliding window scan over a large number of partially overlapped windows and (ii) the limited contextual information (i.e., the size of the patch). More complex models have been more recently proposed to overcome the above drawbacks.

- Tang et al. (2017a) approached minutia extraction as an object-detection problem (similarly to a Faster R-CNN model, Ren et al., 2015), where a Fully Convolutional Network is used to map raw fingerprints to a minutia-score map with 1 position every 16×16 pixels; the positions whose score exceeds a given threshold are candidate minutiae; the neighboring regions are further processed by a second CNN (sharing the same convolutional levels) that also regresses the minutia orientation (see Fig. 3.56).
- Tang et al. (2017b) proposed a unified deep model, denoted as FingerNet, where domain knowledge and end-to-end training are both exploited to improve accuracy. In particular, the model is initialized by converting traditional feature extraction steps (normalization, orientation extraction, enhancement, and minutiae detection) into convolutional layers with fixed weights. Then the above basic layers are expanded with further layers and all the model weights are further tuned during end-to-end training of the model. A composite loss function is used to exploit weak, strong, and ground

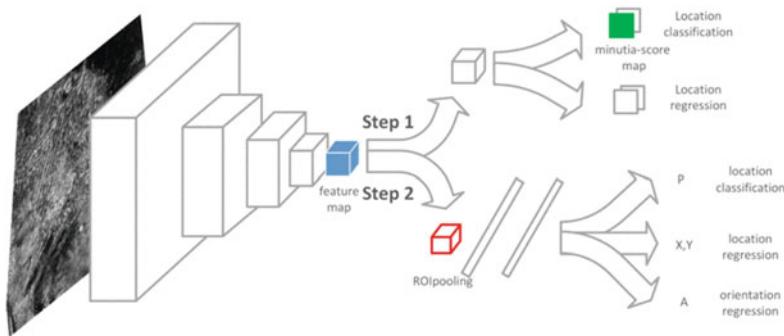


Fig. 3.56 The architecture proposed by Tang et al. (2017b). The proposals generated by the detection branch (Step 1) are classified by a second branch which also calculates their orientations (Step 2). © IEEE. Reprinted, with permission, from Tang et al. (2017b)

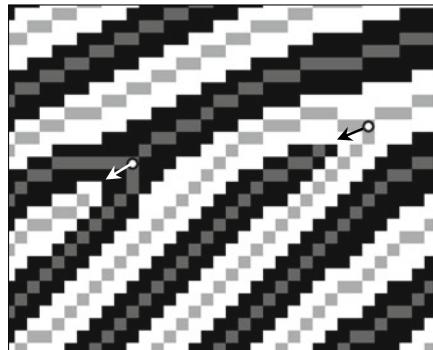
truth labels on orientation, foreground area, and minutiae. The approach by Tang et al. (2017b) is described in more detail and with illustrations in Sect. 6.4.6.

- Another two-step method was proposed by Nguyen et al. (2018b). In the first step, an improved FingerNet (Tang et al. 2017b) is used to produce candidates that are further checked at patch level by a second CNN. While the entire approach is quite complex, its minutiae detection accuracy exceeded previous state-of-the-art methods on FVC2004 and NIST SD27.
- Nguyen et al. (2020) introduced a simple semantic segmentation approach where a light version of the U-Net model (Ronneberger et al. 2015), is used to classify each pixel into 37 classes: 36 classes corresponding to a minutia with a given angle (i.e., with 10 degrees steps) and the last class for non-minutia. Good minutiae detection results are reported for many FVC databases with an inference time of only 130 ms for 640×480 images on GPU.

3.7.4 Minutiae Encoding Standards

Standards have been introduced to define the way minutiae information should be encoded. Most of these standards, including ISO/IEC 19794-2 (2011), ANSI/NIST-ITL 1-2011 (2015), ANSI/INCITS 378 (2004), and CDEFFS (2008), require to encode minutiae data with the same conventions in order to promote interoperability. For example (see Fig. 3.57), ISO/IEC 19794-2 (2011) requires to extract both the ridge and the valley skeletons and to place (i) bifurcation minutiae in correspondence with the ridge skeleton bifurcations (i.e., pixels p with $cn(p) = 3$), and (ii) ridge-ending minutiae in correspondence with the valley skeleton bifurcations (this exploits the ridge-ending/bifurcation

Fig. 3.57 Ridge-ending (black arrow) placement on valley skeleton and bifurcation (white arrow) placement on ridge skeleton according to ISO/IEC 19794-2 (2011). In the figure, ridges are black, valleys are white, and ridge and valley skeletons are gray



duality shown in Fig. 3.6); the minutia direction θ is computed in both the cases (ridge ending and bifurcation) as the bisector of the two angularly closest branches departing from \mathbf{p} .

The main aim of the standards is to achieve interoperability among minutiae templates extracted by different approaches and so, for this purpose, clear and unambiguous rules should be given; unfortunately, none of the current standards is sufficiently clear and exhaustive and new efforts are being made to improve them. The interoperability evaluation conducted in the MINEX (Grother et al., 2006; Wu & Garris, 2007) and MTIT (Bazin & Mansfield, 2007) projects has clearly pointed out such limitations. The main problems are as follows:

- Minutiae position is often defined resorting to the skeleton of a binary fingerprint image, but the skeleton itself depends on enhancement, binarization, and thinning algorithms.
- While criteria are given to validate minutiae and filter out spurious minutiae (e.g., if one of the branches departing from a minutia cannot be tracked for more than a given length, then the minutia is invalid), an iterative unordered application of these rules can also lead to unintended deletion of valid minutiae.
- No specific rules are given to define how to deal with very noisy regions and with high curvature regions (i.e., singularities) where minutiae tend to cluster.

Bolle et al. (2002) addressed the problem of precisely and consistently locating the minutiae points in the gray-scale fingerprint pattern. In fact, different feature extraction algorithms tend to locate the minutiae at slightly different positions (depending on their operational definition of minutia and the intermediate processing steps) and this may lead to interoperability problems. Bolle et al. (2002) provided a formal definition of minutia based on the gray-scale image that allows the location and orientation of an existing minutia to be more precisely determined. However, this approach has been ignored by current standards that define the minutiae location on the basis of binary skeletons.

3.7.5 Benchmarking Minutiae Extraction

Since minutiae extraction is a fundamental fingerprint processing step, benchmarking it in isolation (instead of when it is embedded in a whole fingerprint recognition system) can be very useful to assess the performance of existing techniques and guide future developments. One of the main obstacles is the need for accurate ground truth information. In fact, manual labeling not only is boring and prone to errors, but it is very critical in low-quality fingerprints that, on the other hand, constitute the most interesting cases. Synthetic fingerprint generation is a valuable strategy to obtain reliable ground truth labels also for low-quality data (see Chap. 7).

Some studies have been published where minutiae extraction techniques (both commercial solutions and academic developments) have been tested on available datasets:

- Kayaoglu et al. (2013) used a GUI tool to manually mark a total of 116,000 minutiae over four FVC datasets (acquired with optical, capacitive, and thermal sensors). Commercial minutiae extractors were compared with manually extracted minutiae in terms of associated fingerprint recognition accuracy to conclude that manual extraction leads to significantly better performance especially on low-quality fingerprints.
- Chugh et al. (2017) used the four mentioned labeled FVC datasets in conjunction with NIST SD27 consisting of 257 rolled fingerprints whose minutiae were marked by forensic examiners. The test also includes a corpus of synthetically generated fingerprints. Three COTS algorithms and an open-source solution (*mindtct* by NIST) have been compared in terms of missing and spurious minutiae, also by using the overall Goodness Index metric introduced by Ratha et al. (1995). In the experiments, the fingerprints were divided into five bins according to their NFIQ2.0⁶ quality: 0–20, 21–40, 41–60, 61–80, and 81–100; the best algorithm produced a percentage of spurious minutiae ranging from 44% (bin 0–20) to 10% (bin 81–100) and a percentage of missing minutiae ranging from 29% (bin 0–20) to 16% (bin 81–100). The open-source algorithm extracted a significantly higher number of spurious minutiae across all fingerprint quality. Positional and angular errors were also measured for paired minutiae: typical positional errors are 3, ..., 5 pixels and angular errors are 5°, ..., 8°.
- The impact of different types of minutiae detection errors constitutes the subject of the study by Grosz et al. (2020). Their evaluations revealed that the performance of fingerprint minutiae matchers is more susceptible to non-linear distortion and missing minutiae than spurious minutiae and small positional displacements of the minutiae locations.

⁶ NFIQ2.0 is introduced in Sect. 3.11.3.

3.8 Minutiae Filtering

A post-processing stage is often useful in removing the spurious minutiae detected in highly corrupted fingerprint regions or introduced by preceding processing steps (e.g., thinning). However, some of the techniques introduced in Sect. 3.7 (e.g., especially learning-based methods working on large input neighborhoods) do not need a separate post-filtering stage.

Two main post-processing types have been proposed:

- Structural post-processing.
- Minutiae filtering in the gray-scale domain.

3.8.1 Structural Post-Processing

Simple structural rules may be used to detect many of the false minutiae that usually affect thinned binary fingerprint images. Xiao and Raafat (1991) identified the most common false-minutiae structures and introduced an ad hoc approach to remove them (Fig. 3.58). Their algorithm is rule-based and requires some numerical characteristics associated with the minutiae as input: the length of the associated ridge(s), the minutia angle, and the number of facing minutiae in a neighborhood. As shown in Fig. 3.58, the algorithm connects facing endpoints (a) and (b); removes bifurcations facing with endpoints (c) or with other bifurcations (d); and removes spurs (e), bridges (f), triangles (g), and ladder structures (h).

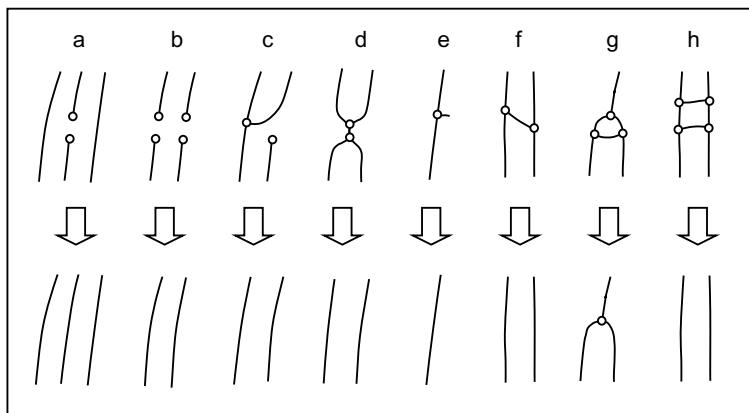


Fig. 3.58 The most common false-minutiae structures (top row) and the structural changes resulting from their removal (bottom row)

Hung (1993) and Zhao and Tang (2007) exploited the minutiae duality (Fig. 3.6) to purify false minutiae extracted from binary thinned images.

- In Hung (1993), both ridge and valley skeletons are extracted and only ridge minutiae having a counterpart (of complementary type) in the valley skeleton are retained. A graph is defined for both ridge and valley skeletons by assigning a vertex to each ridge ending and bifurcation and by assigning an edge to each ridge. Each edge is characterized by the length of the corresponding ridge, and the degree of a vertex is given by the number of converging edges. Spurs (i.e., very short edges) and holes (i.e., loops with a very small diameter) are first removed by considering some property of the ridge graph. Bridges between adjacent ridges are then removed by exploiting their relation with breaks in the dual space.
- Zhao and Tang (2007) argue that for most of the false minutiae there is at least a bridge structure; referring to Fig. 3.58, this is true not only for bridges (f), triangles (g), and ladders (h) in the ridge skeleton but also for breaks (a) in the valley skeleton. An H-point is defined as a bridge structure in one of the two (ridge or valley) skeletons and its corresponding break in the dual skeleton; a simple rule is then defined to detect and remove H-points thereby eliminating many false minutiae. However, to avoid cancellation of genuine structures, the H-point removal must be embedded within an ordered sequence of deletion steps: short breaks, spurs, H-points, close minutiae, and border minutiae.

In the approach by Farina et al. (1999), spurs and bridges are removed based on the observation that in a “spurious” bifurcation, only two branches are generally aligned whereas the third one is almost orthogonal to the other two. Short ridges are removed on the basis of the relationship between the ridge length and the average distance between ridges. Ridge endings and bifurcations are then topologically validated; they are (i) removed if topological requirements are not satisfied; (ii) classified as less reliable if the requirements are not fully satisfied; and (iii) considered as highly reliable minutiae, otherwise. An example is shown in Fig. 3.59.

A slightly different implementation of spurious minutiae removal was proposed by Kim et al. (2001). In their work, local orientation and flow of ridges are key factors for post-processing to avoid eliminating true minutiae. Bhowmick et al. (2002) assign a score to each minutia based on clarity of ridge and valley flow and the noise level in the locality of the minutia; Kim (2005) and Chen et al. (2007) also assign scores to the minutiae based on local ridge connectivity, interspacing, and symmetry. The scores can then be used either for immediate filtering or minutiae weighting during matching.

The filtering method of Bhanu et al. (2000) verifies each minutia, detected from the thinned binary image, through correlation with logical templates (i.e., template matching) adapted to the local ridge orientation. In Bhanu and Tan (2001), an evolution of the above

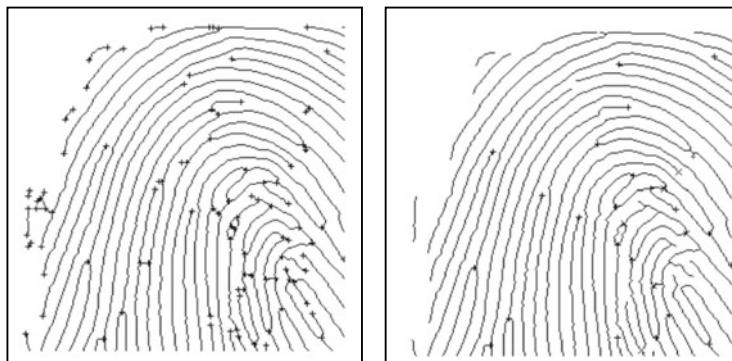


Fig. 3.59 Minutiae post-processing according to Farina et al. (1999). On the right, most of the false minutiae present in the thinned binary image (on the left) have been removed. © Elsevier. Reprinted, with permission, from Farina et al. (1999)

method is proposed where templates are not static, but are learned in a supervised manner from examples.

3.8.2 Minutiae Filtering in the Gray-Scale Domain

A direct gray-scale minutiae filtering technique reexamines the gray-scale image in the spatial neighborhood of a detected minutia with the aim of verifying the presence of a real minutia.

Maio and Maltoni (1998b) used a three-layer neural network where a partial weight sharing allows the ridge-ending/bifurcation duality to be exploited (Fig. 3.60). The minutiae neighborhoods in the original gray-scale image are normalized, with respect to their angle and the local ridge frequency, before passing them to a neural network classifier, which classifies them as ridge ending, bifurcation, and non-minutia. Figure 3.61b shows the same minutiae neighborhoods of Fig. 3.61a after the normalization. To take advantage of the ridge-ending/bifurcation duality, both the original neighborhood and its negative version constitute the input to the neural network classifier. Additionally, to avoid the problems related to training large networks, the dimensionality of the normalized neighborhoods is reduced through the Karhunen–Loeve transform (Jolliffe, 1986). Experimental results showed that this filtering method, in spite of a certain increase in missed minutiae, provides a significant reduction in false minutiae and misclassified minutiae (i.e., a ridge ending detected as bifurcation and vice versa) errors.

The minutiae verifier of Prabhakar et al. (2003) operates on the gray-scale neighborhoods extracted from the original image after enhancement through the Gabor filtering (Hong et al., 1998). Minutiae neighborhoods are normalized with respect to minutiae

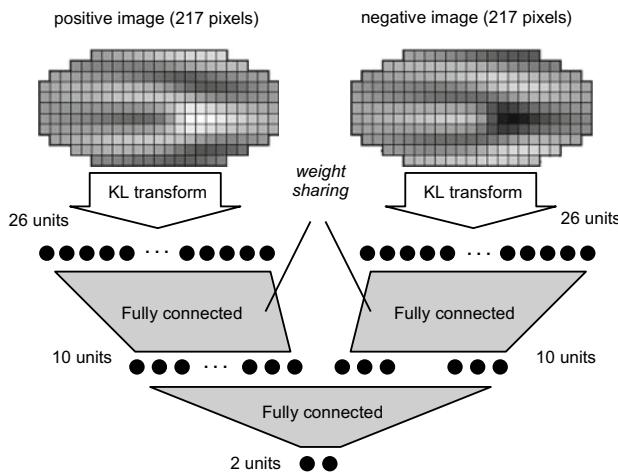


Fig. 3.60 The neural network architecture to classify gray-scale minutiae neighborhoods into ridge ending, bifurcation, and non-minutiae (Maio & Maltoni, 1998b). © IEEE. Reprinted, with permission, from Maio and Maltoni (1998b)

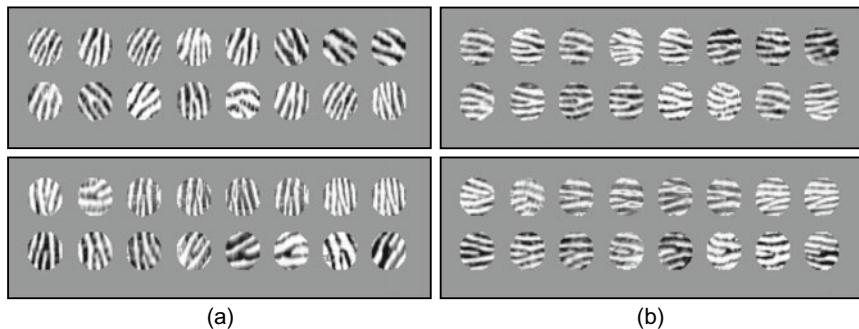


Fig. 3.61 **a** Minutiae neighborhoods (ridge-ending minutiae at the top, bifurcation minutiae at the bottom) as they appear in the original gray-scale images; **b** the same neighborhoods have been normalized with respect to minutiae angle and local ridge frequency (Maio & Maltoni, 1998b). © IEEE. Reprinted, with permission, from Maio and Maltoni (1998b)

angle and local ridge frequency. The resulting patterns are classified through a Learning Vector Quantizer (Kohonen et al., 1992) trained in a supervised fashion to discriminate between minutiae and non-minutiae. The authors obtained a classification accuracy of 87% and a reduction of about 4% fingerprint matching error when their minutiae verification algorithm was embedded into the minutiae-based fingerprint verification system described in Jain et al. (1997).

Chikkerur et al. (2005) proposed two minutiae verifiers:

- The first one is based on the response of the minutiae neighborhood to a bank of steerable wedge filters. The response is fed to a feedforward backpropagation network to classify the inputs as either minutiae or non-minutiae neighborhoods.
- The second (and more accurate) one encodes the minutiae neighborhoods as a linear sum of basis images made up of multi-resolution Gabor elementary functions. A parametric Bayesian classification is then applied. The authors report a minutiae verification accuracy of 98%.

Most of the deep learning techniques introduced in Sect. 3.7.3 can be used for minutiae filtering in the gray-scale domain. In particular,

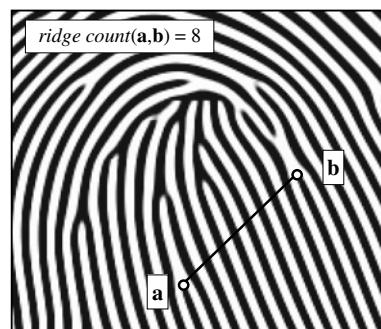
- the patch-based classifiers (e.g., Sankaran et al. 2014; Jiang et al. 2016; Darlow and Rosman 2017), can be directly used for minutiae post-filtering without any sliding window scan;
- in the two-stage methods (e.g., Tang et al. 2017b; Nguyen et al., 2018b), the second stage can be conceived as a minutiae post-filtering.

3.9 Estimation of Ridge Count

Absolute position, direction, and type of minutiae (e.g., ridge ending or bifurcation) are not the only features that may be used for fingerprint recognition. In fact, forensic experts and latent fingerprint examiners have often used *ridge count* to increase the reliability of their analysis (Henry, 1900). Ridge count is an abstract measurement of the distances between any two points in a fingerprint image (Lin & Dubes, 1983). Let **a** and **b** be two points in a fingerprint; then the ridge count between **a** and **b** is the number of ridges intersected by segment **ab** (Fig. 3.62).

Ridge count has been typically used in forensic matching because of the difficulty of human experts to work in the Euclidean space. However, because the early automated

Fig. 3.62 In this example, the number of ridges intersected by segment **ab** (ridge count between **a** and **b**) is 8.



fingerprint identification systems (AFIS) were developed from an intuitive design geared toward duplicating the performance of human experts in matching fingerprints, ridge counts have been used in the AFIS systems. With an increased interest in improving the performance of fingerprint recognition systems in commercial applications, several authors have proposed ridge counts as features.

Although the general definition of ridge count includes measuring the number of ridges between any two points (**a** and **b** in Fig. 3.62) in the fingerprint images, typically, these points coincide with some well-defined points in the fingerprint pattern (e.g., position of the singularities or position of the minutiae). For example, in forensic applications of AFIS, it is common to count the ridges between core and delta.

There exist two main approaches for counting the number of ridges between points **a** and **b** in a fingerprint image:

- Determine the number of (0–1) transitions along the segment **ab** in a binarized image.
- Determine the number of local maxima in the section **ab** of a gray-scale image. Refer to the *x*-signature method (Fig. 3.21) for an example of a possible implementation.

In both cases, the estimation of ridge count may be problematic in noisy areas, near singularities, near minutiae, and when the segment orientation is close to the underlying ridge orientation. Sha et al. (2006) suggest using only a subset of highly reliable ridge counts and propose a labeling-based algorithm for their extraction. Kovacs-Vajna (2000) matched the ridge profiles between pairs of minutiae using the dynamic time warping technique. This method is conceptually similar to ridge counting, even if it cannot be directly used to estimate the ridge count but only to verify that the ridge–valley profiles between two pairs of corresponding points are similar.

3.10 Pore Detection

While pores are highly distinctive and extremely important for latent fingerprint examiners, their practical utility to improve automated fingerprint recognition algorithms is questionable (Zhang & Jain, 2010). More details are provided in Chap. 4. Noticing that a single ridge with a string of connected pores may be incorrectly extracted as two ridges, Hara (2011a) proposed an algorithm to extract and then eliminate pores to avoid such a problem. Given sufficiently high-resolution fingerprint images (e.g., 1,000 dpi), several techniques can be used for automatic pore extraction.

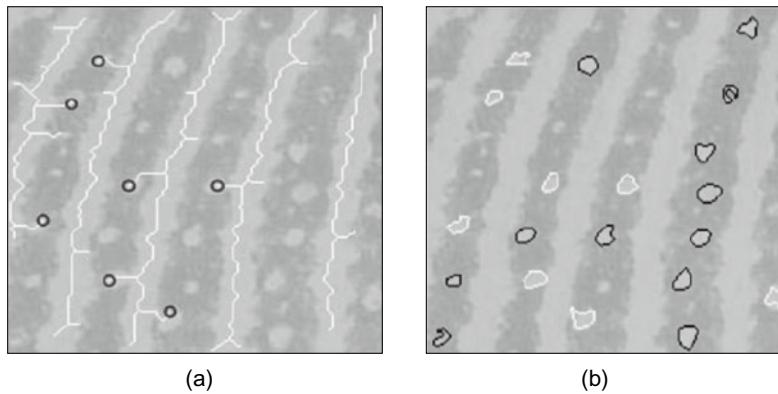


Fig. 3.63 Examples from Stosz and Alyea (1994): **a** detection of open pores from the skeleton; **b** extraction of open pores (white) and closed pores (black). Images courtesy of K. Kryszczuk

3.10.1 Skeletonization

One of the first methods was proposed by Stosz and Alyea (1994). Specifically, the locations of all endpoints (with at most one neighbor) and branch points (with exactly three neighbors) in the skeleton image are extracted and each endpoint is used as a starting location for tracking the skeleton. The tracking algorithm advances one pixel at a time until one of the following stopping criteria is encountered: (1) another endpoint is detected, (2) a branch point is detected, and (3) the path length exceeds a maximum allowed value. Condition (1) implies that the tracked segment is a *closed pore*, while condition (2) implies an *open pore*. Finally, skeleton artifacts resulting from scars and wrinkles are corrected and pores from reconnected skeletons are removed. An example of pore extraction is shown in Fig. 3.63. Another skeletonization-based pore detection algorithm with more anatomical constraints with respect to the Stosz and Alyea (1994) algorithm was proposed by Kryszczuk et al. (2004).

3.10.2 Filtering

Jain et al. (2007) noted that skeletonization is effective for pore extraction only when the image quality is very good and the image resolution is very high. Hence, unlike in previous studies, pores were extracted using the Gabor filters and Mexican hat wavelet transform; see Fig. 3.64 for some details of a pore extraction process. Other filtering-based approaches were introduced by Abhyankar and Schuckers (2010) and Zhao et al. (2010a); the former is still based on wavelet transform while the latter relies on DoG (Difference of Gaussians) computed at various scales.

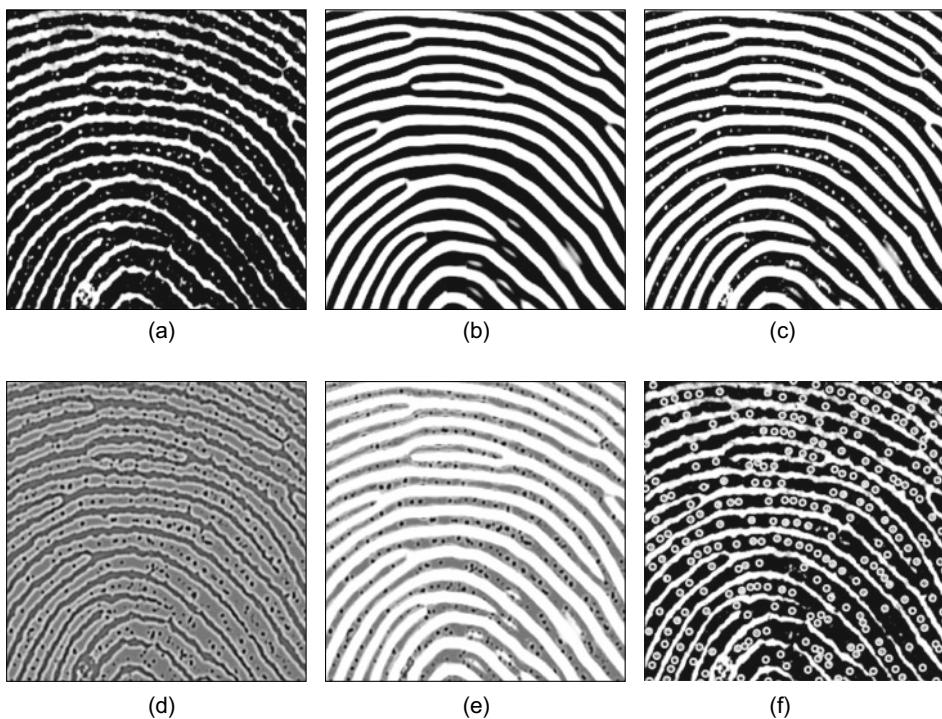


Fig. 3.64 Pore extraction in Jain et al. (2007): **a** a partial fingerprint image at 1,000 dpi; **b** enhancement of the image shown in **(a)** using a Gabor filter-based contextual technique; **c** a linear combination of **(a)** and **(b)**; **d** wavelet bandpass filtering of the image in **(a)** that exhibits small dark blob in correspondence of the pores; **e** a linear combination of **(d)** and **(b)**, and **f** extracted pores (white circles) after thresholding the image in **(e)**. © IEEE. Reprinted, with permission, from Jain et al. (2007)

Zhao et al. (2010b) argued that an anisotropic filtering is preferable because open pores are not circularly shaped and their response to isotropic filtering can be weak. The proposed adaptive anisotropic filtering approach achieved a true detection rate of 84.8% and a false detection rate of 17.6% on a database of 24 fingerprint images acquired at 1,000 dpi.

3.10.3 Topological Approaches

In the method proposed by Malathi et al. (2010), pores detection is performed by employing a topological algorithm known as marker controlled watershed segmentation. The method is claimed to be able to extract pores also on 500 dpi images.

The Teixeira and Leite (2013) method relies on a multi-scale morphological operator used in conjunction with local orientation to extract both closed and open pores. The accuracy reported on the 30 pore-annotated fingerprints included in the PolyU HRF Database is higher than previous filtering-based methods. A post-processing technique was then proposed by Teixeira and Leite (2014) to reduce the number of false pore detection; the filtering is based on the observation that distance between adjacent pores in a ridge is proportional to the ridge width.

3.10.4 Deep Learning Methods

In the method introduced by Donida Labati et al. (2018), the fingerprint image is first processed by a five-layer CNN aimed at enhancing pores while discarding the ridge–valley pattern. This can be seen as a sort of image-to-image translation. Pore centroids are extracted from the obtained pore map by looking at connected components after simple binarization. A second CNN is then used to check each candidate pore by also taking into account the ridge map and some symmetry information. In their experiments, the authors also considered touchless reader-acquired fingerprints and latent fingerprints and, as expected, confirmed that in these cases pore extraction is very challenging. An improved version of this method was introduced by Jang et al. (2017), relying on a deeper CNN model and a more accurate selection of pores from the pore map. More complex and still deeper models (e.g., ResNet and DenseNet) were adopted by other researchers (Anand & Kanhangad, 2019; Shen et al., 2019) to gain minor improvements over the previous methods. In Ding et al. (2021), a modified U-Net is used to extract sweat glands and sweat pores from OCT volumes.

3.11 Estimation of Fingerprint Quality

Many different meanings could be associated with the term quality. However, most of the researchers have focused only on the *operational* definition of quality, that is, the estimate of quality must be predictive of the utility of features and/or “matchability” of the fingerprint images.

In general, fingerprint quality can be estimated at a *global* level (i.e., a single quality value is derived for the whole image) or at a *local* level (i.e., a distinct value is estimated for each block/pixel of the image). Of course, the availability of local estimates is preferable since it is more descriptive, and in any case, one could compute the global quality from the statistics of the local estimates. Estimating fingerprint quality is important since it has the following properties (Grother & Tabassi, 2007):

- Rejects very low-quality samples (images) during enrollment or/and selects the best sample(s).
- Isolates unrecoverable regions (see Sect. 3.6) where fingerprint enhancement is counterproductive as it leads to the detection of several spurious features.
- Adapts the matching strategy to the quality of fingerprints.
- Assigns weights to features (at the matching stage) according to the quality.

The most popular approach to estimate global fingerprint quality is NFIQ (NIST Fingerprint Image Quality). The first version of NFIQ dates back to 2004 (Tabassi et al., 2004) and the second version, known as NFIQ 2.0, was introduced one decade later (Tabassi et al., 2021). NFIQ defines the quality as a prediction of a matcher performance: good-quality fingerprints are likely to produce high match scores. An important advantage of this method is that it does not require a ground truth provided by a human expert; in fact, defining the ground truth by visual inspection is quite complicated, could lead to subjective evaluations, and is not necessarily the best approach when the focus is on automated matching algorithms.

NFIQ and NFIQ 2.0 were conceived for plain and rolled fingerprint images at 500 dpi captured using optical sensors or scanned from inked cards. Some adaptations (at least retraining and calibration) would be necessary for other categories of sensors. Some authors introduced specific quality metrics for latent fingerprints (Sankaran et al., 2013; Yoon et al., 2013) and for high-resolution fingerprints (e.g., 1000 dpi) where pore visibility and reliability are quality-related (Zhao et al., 2010c; Teixeira & Leite, 2017). Quality estimation of latent fingerprints is discussed in more detail in Chap. 6.

Some studies (Young & Elliott, 2007) have shown that, on average, (i) fingerprint images from the index and middle fingers exhibit better quality than images taken from the ring and little fingers, and (ii) Whorl is the fingerprint class containing the largest proportion of good-quality fingerprint images, whereas Arch is at the opposite side of the quality scale. This knowledge, when available, could be useful as a prior probability to quality estimation approaches.

In the following, a brief summary of the existing global and local approaches for estimating fingerprint image quality is reported. More comprehensive (and comparative) reviews can be found in Alonso-Fernandez et al. (2007) and Yao et al. (2018).

3.11.1 Local Quality Estimation

A number of methods have been proposed to estimate block-wise fingerprint quality. Most of them estimate the local quality according to the local orientation reliability (see Sect. 3.3.1): Shen et al. (2001), Lim et al. (2002), Chen et al. (2004a), Yao et al. (2004), Chen et al. (2005), Zhu et al. (2005), and Panetta et al. (2019). The way the reliability of a single orientation is obtained depends on the orientation computation method, but is

usually related to the coherence of a set of orientation estimations in a given neighborhood (see Eq. (3.4) and Fig. 3.14c). Although orientation coherence is a very powerful feature to measure quality, it fails near the singularities; in fact, singularities are characterized by high curvatures which result in low coherence (see Fig. 3.28). The use of linear and parabolic symmetry operators (see Fronthaler et al. 2006, 2008a, b) overcomes this problem. In fact, the orientations in good-quality regions highly correlate with (i) linear symmetry filters outside singularities and (ii) parabolic symmetry filters near the singularities, whereas low-quality regions weakly respond to both types of filters.

Other features can be used to characterize local quality:

- Statistics derived from pixel intensities (e.g., mean, variance, contrast, gradient magnitude, histogram properties, etc.) as proposed by Shi et al. (2004), Uchida (2004), Chen et al. (2004b), Lim et al. (2004), Hwang (2004), Qi et al. (2005a), and Zhu et al. (2005).
- Ridge frequency, ridge thickness, and ridge-to-valley thickness (Lim et al., 2002; Zhu et al., 2005); a substantial deviation of these values from their typical range is alarming and can be exploited to detect unreliable regions.
- The presence of dominant frequencies in the Fourier spectrum of a local region (Lim et al., 2004) and the local resemblance to an ideal sinusoidal wave pattern in the spatial domain (Lee et al., 2006).
- The regularity of the manifold determined by fingerprint patches in a local neighborhood (Tao et al., 2012).

Liu et al. (2020) performed a cluster analysis of a large number of fingerprint blocks to group them into four quality classes (further expanded with background and singular regions) and then trained a simple neural network classifier to compute the block class starting from a feature vector including some of the previously discussed local features.

It is difficult to compare the relative performance of these methods quantitatively because of a non-uniform quality definition. In ANSI/NIST-ITL 1-2011 (2015) standard, the local quality of fingerprint is defined according to the reliability of different level features (six levels). However, there is no public fingerprint database based on this standard for scientific research.

3.11.2 Global Quality Estimation

Qi et al. (2005b) combine local and global features already available in the literature, but among the global features, the authors suggest taking into account the size of the foreground area, the foreground centered with respect to the image center, and the presence of detectable singularities.

A good-quality fingerprint image exhibits a ring around the origin of the frequency coordinate in the Fourier spectrum, because the ridge–valley patterns are quasi-periodic structures and present a dominant frequency in most directions with an almost uniform modulus. The implementation of ad hoc ring detectors allows an estimate of the overall fingerprint image quality: see Uchida (2004), Chen et al. (2005), and Lee et al. (2005).

Some authors (Munir et al., 2012; Sharma & Dey, 2019) proposed hierarchical or fuzzy clustering to group fingerprints in quality classes and in particular to infer their appearance: *normal*, *wet*, or *dry*.

An interesting quality measure was introduced by Yao et al. (2015): this metric only relies on the minutiae points and therefore it can be computed starting from templates (e.g., in the ISO standard format) with no need of the original fingerprint image. To compute the quality, the set of minutiae is modeled with the convex-hull and Delaunay triangulation. The authors observed that bad-quality samples generate tiny and extremely narrow triangles (considered as unreasonable) due to spurious minutiae. Hence, geometrical characterization of minutiae triangles was used to determine the quality. Comparative evaluations in Yao et al. (2018) denote that, in spite of its simplicity, this approach is quite effective when used to select the best sample for enrollment.

The size of the overlapping area of two fingerprints has a large impact on matching performance. To ensure a large overlapping area, the central area of fingerprints (corresponding to the most distinctive region) is preferred to be captured in enrollment and recognition stages. Thus, a fingerprint with a large central area should be assigned a higher quality than a fingerprint, which has a similar image quality and overall size, but a smaller central area. To take this factor into account, Hara (2011b) proposed to estimate fingerprint quality based on completeness of central area (called pattern area) and ridge quality in this area.

To evaluate the effectiveness of global quality estimation approaches, Grother and Tabassi (2007) proposed to measure the improvement in matching accuracy when rejecting low-quality fingerprints. For this purpose, the FNMR can be plotted vs increasing amounts of rejections (see Fig. 3.65): curves closer to the origin denote more effective quality estimation algorithms. A mix of fingerprint feature extraction and matching techniques should be considered to avoid that results are biased by a specific algorithm.

3.11.3 NFIQ (NIST Fingerprint Image Quality)

NFIQ 1.0

The quality $q(x_i)$ of fingerprint image x_i is defined as the prediction of its normalized matching score $\text{normscore}(x_i)$. Let T be a training set containing n fingers and a pair of image samples (x_i, x'_i) for each finger $i = 1, \dots, n$. The normalized matching score of x_i is defined according to the separation of the x_i genuine matching score from the x_i impostor scores:

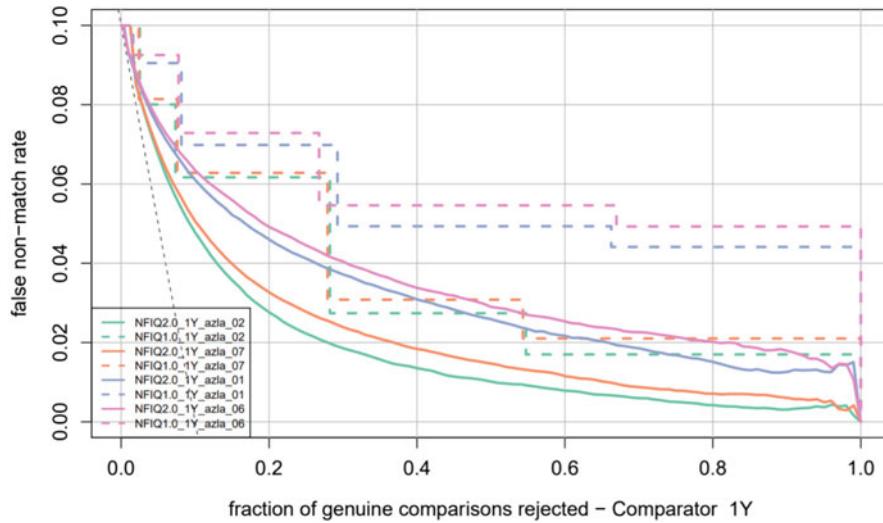


Fig. 3.65 In this graph, extracted from Tabassi et al. (2021), FNMR is plotted for different amounts of rejections. Fingerprints are sorted by quality so that the first samples rejected are the lowest quality ones. The solid green curve corresponding to an NFIQ 2.0 approach is here the most effective. Images courtesy of E. Tabassi

$$\text{normscore}(x_i) = \frac{\text{score}(x_i, x'_i) - \text{avg}_{j=1 \dots n, j \neq i}(\text{score}(x_i, x_j))}{\text{stdev}_{j=1 \dots n, j \neq i}(\text{score}(x_i, x_j))},$$

where $\text{score}(a, b)$ returns the matching score between the two fingerprints a and b according to a given automatic matcher, $\text{avg}()$ is the average value, and $\text{stdev}()$ the standard deviation of the scores.

Given a feature vector \mathbf{v}_i extracted from x_i , a mapping between \mathbf{v}_i and $q(x_i)$ can be found by regression over T by considering the pairs $\langle \mathbf{v}_i, \text{normscore}(x_i) \rangle$, $i = 1 \dots n$. Tabassi et al. (2004) preferred to formulate the problem as a classification problem (instead of as a regression problem) in order to quantize the fingerprint quality into just five values; for this purpose, a neural network classifier is trained to classify the feature vector into one of the five predefined quality classes (where class one means top quality and class five the worst quality). Each feature vector is constituted by 11 features. To extract these features, the method calculates a quality map of the foreground according to the consistency of local orientation, the local contrast, and the curvature. Minutiae detection is then performed and the reliability of each detected minutiae point is computed according to simple pixel intensity statistics (mean and standard deviation) within the immediate neighborhood of the minutiae point. The minutiae reliability is then combined with the local quality at the minutiae location (from the quality map) to produce a quality measure for each minutiae.

NFIQ 2.0

NFIQ 2.0, the natural replacement of NFIQ, is described in Tabassi et al. (2021). The main desiderata considered during its development were: better efficiency, finer granularity (101 quality bins instead of 5), and modular design. A comprehensive literature survey was initially performed to select 155 candidate features: 14 features were finally chosen according to their predictive power, computation time, and correlation (see Table 3.2). Feature vectors built by concatenating the above features were then used to train a Random Forest classifier that predicts the fingerprint quality.

A lite version of NFIQ2.0 was proposed by Tabassi et al. (2013). This method is based on the Bag-of-Words approach and can be efficiently run on low-resource clients. A large training set of blocks (e.g., 24×24 pixels) extracted from fingerprint images are clustered by spatial similarity to form a codebook (or visual dictionary). For this purpose, the authors used a self-organizing map (SOM) instead of a classical K-means approach. One of the reasons is that the topological ordering resulting from SOM clustering allows to speed up the search of the Best Matching Unit (BMU) during the online phase. Given a codebook, a fixed-size feature vector can be extracted from a fingerprint image by (i) subdividing it into blocks; (ii) searching the BMU in the codebook for each block, and (iii) constructing an histogram of the BMU indices. A random forest classifier is then trained to map feature vectors to quality bins.

Table 3.2 Features used in NFIQ2.0. To build fixed-length feature vectors, the local features (estimated block-wise) are aggregated by computing the mean, standard deviation, and histogram. NFIQ2.0 quality features are standardized as part of ISO/IEC 29794-4 (2017) and serve as the reference implementation of the standard

Number	Type	Feature
1	Local	FDA: Frequency-Domain Analysis
2	Local	LCS: Local Clarity Score
3	Local	OCS: Orientation Certainty Level
4	Local	OFL: Orientation Flow
5	Local	RVU: Ridge–Valley Uniformity
6	Global	MU: Mean Gray Level
7	Global	MMB: Mean of Blocks Gray-Level Mean
8	Global	Minutiae Count (detected by FingerJetFX)
9	Global	Minutiae Count in the Central Region (detected by FingerJetFX)
10	Global	The Mean of Minutiae Quality (based on gray-level stats)
11	Global	The Mean of Minutiae Quality (based on orientation coherence)
12	Global	ROI Area Mean
13	Global	ROI Orientation Map Coherence Sum
14	Global	ROI Relative Orientation Map Coherence Sum

3.12 Summary

Most of the early work in fingerprint analysis was based on general-purpose image processing techniques, and therefore, the resulting matching performance was often inadequate. Subsequently, several special-purpose (domain-specific) algorithms have been designed for processing fingerprint images and exploiting the peculiarity of fingerprint patterns; an example is a contextual enhancement. In the last decade, machine learning approaches have been applied to fingerprint processing and feature extraction, initially based on dictionaries and bag-of-words approaches and more recently relying on successful deep learning models such as CNN. This allowed to extract more reliable fingerprint representations especially in very poor-quality fingerprints. In particular, fully automated processing of latent fingerprints (i.e., including segmentation and minutiae extraction) has been made possible by designing techniques leveraging both learned models and domain knowledge.

Today, fingerprint image processing and feature extraction are undoubtedly a mature field, and effective algorithms are available for most of the problems: local orientation and local frequency estimation, foreground segmentation, image enhancement, singularity detection, minutiae, pore extraction, etc. Automatic processing of latent fingerprints is also at hand, even if here we can expect a further improvement in terms of accuracy and efficiency.

Deep learning approaches often achieve state-of-the-art accuracy on many of the above tasks, but usually, this comes at the cost of computing intensive algorithm running on a GPU. Many low-resource clients or embedded devices do not have such processing facilities, but fortunately, traditional approaches work well enough for most commercial applications. For example, a recent benchmarking on different enhancement techniques confirms that contextual filtering based on the Gabor filters is still one of the most effective algorithms (Schuch et al., 2018b).

References

- Abhyankar, A., & Schuckers, S. (2010). Towards integrating level-3 features with perspiration pattern for robust fingerprint recognition. In *2010 IEEE International Conference on Image Processing* (pp. 3085–3088). <https://doi.org/10.1109/ICIP.2010.5654261>.
- Almansa, A., & Lindeberg, T. (1997). Enhancement of fingerprint images using shape-adapted scale-space operators. In J. Sporring, M. Nielsen, L. Florack & P. Johansen (Eds.), *Gaussian Scale-space Theory* (pp. 21–30). Kluwer.
- Almansa, A., & Lindeberg, T. (2000). Fingerprint enhancement by shape adaptation of scale-space operators with automatic scale selection. *IEEE Transactions on Image Processing*, 9(12), 2027–2042.
- Alonso-Fernandez, F., Fierrez-Aguilar, J., & Ortega-Garcia, J. (2005). An enhanced Gabor filter-based segmentation algorithm for fingerprint recognition systems. In *Proceedings of International Symposium on Image and Signal Processing and Analysis*.

- Alonso-Fernandez, F., Fierrez, J., Ortega-Garcia, J., Gonzalez-Rodriguez, J., Fronthaler, H., Kollreider, K., & Bigun, J. (2007). A comparative study of fingerprint image-quality estimation methods. *IEEE Transactions on Information Forensics and Security*, 2(4), 734–743.
- Anand, V., & Kanhangad, V. (2019). Pore detection in high-resolution fingerprint images using deep residual network. *Journal of Electronic Imaging*, 28(2), 020502.
- ANSI/INCITS. (2004). INCITS 378-2004—Finger minutiae format for data interchange. ANSI/INCITS standard.
- ANSI/NIST—CDEFFS group. (2008). Data format for the interchange of extended fingerprint and palmprint features—Addendum to ANSI/NIST-ITL 1-2007. ANSI/NIST, Working Draft 0.2. <http://www.fingerprint.nist.gov/standard/cdeffs>. Accessed 27 Nov 2008.
- Araque, J. L., Baena, M., Chalela, B. E., Navarro, D., & Vizcaya P. R. (2002). Synthesis of finger print images. In *Proceedings of 16th International Conference on Pattern Recognition* (Vol. 2, pp. 442–445).
- Arcelli, C., & Baja, G. S. D. (1984). A width independent fast thinning algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(7), 463–474.
- Areekul, V., Watchareeruetai, U., Suppasriwasuseth, K., & Tantaratana, S. (2005). Separable Gabor filter realization for fast fingerprint enhancement. In *Proceedings of International Conference on Image Processing* (Vol. 3, pp. 253–256).
- Areekul, V., Suppasriwasuseth, K., & Jirachawang, S. (2006). The new focal point localization algorithm for fingerprint registration. In *Proceedings of 18th International Conference on Pattern Recognition* (Vol. 4, pp. 497–500).
- Ashbaugh, D. R. (1999). *Quantitative–qualitative friction ridge analysis: An introduction to basic and advanced ridgeology*. CRC Press.
- Aujol, J. F., Gilboa, G., Chan, T., & Osher, S. (2006). Structure-texture image decomposition-modeling, algorithms, and parameter selection. *International Journal of Computer Vision*, 67, 111–136.
- Ballard, D. H. (1981). Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 3(2), 110–122.
- Bartunek, J. S., Nilsson, M., Sallberg, B., & Claesson, I. (2013). Adaptive fingerprint image enhancement with emphasis on preprocessing of data. *IEEE Transactions on Image Processing*, 22(2), 644–656.
- Baruch, O. (1988). Line thinning by line following. *Pattern Recognition Letters*, 8(4), 271–276.
- Bazen, A. M., & Gerez, S. H. (2001). Segmentation of fingerprint images. In *Proceedings of Workshop on Circuits Systems and Signal Processing*.
- Bazen, A. M., & Gerez, S. H. (2002). Systematic methods for the computation of the directional fields and singular points of fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7), 905–919.
- Bazin, A. I., & Mansfield, T. (2007). An investigation of minutiae template interoperability. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 13–18).
- Bernard, S., Boujemaa, N., Vitale, D., & Bricot, C. (2002). Fingerprint segmentation using the phase of multiscale Gabor wavelets. In *Proceedings of Asian Conference Computer Vision*.
- Bhanu, B., Boshra, M., & Tan, X. (2000). Logical templates for feature extraction in fingerprint images. In *Proceedings of 15th International Conference on Pattern Recognition* (Vol. 2, pp. 850–854).
- Bhanu, B., & Tan, X. (2001). Learned templates for feature extraction in fingerprint images. In *Proceedings of Conference on Computer Vision and Pattern Recognition* (Vol. 2, pp. 591–596).
- Bhowmick, P., Bishnu, A., Bhattacharya, B. B., Kundu, M. K., Murthy, C. A., & Acharya, T. (2002). Determination of minutiae scores for fingerprint image applications. In *Proceedings of Indian Conference on Computer Vision Graphics Image Processing* (pp. 463–468).

- Bhowmick, P., & Bhattacharya, B. B. (2009). Removal of digitization errors in fingerprint ridgelines using B-splines. *Pattern Recognition*, 42(3), 465–474.
- Bian, W., Luo, Y., Xu, D., & Yu, Q. (2014). Fingerprint ridge orientation field reconstruction using the best quadratic approximation by orthogonal polynomials in two discrete variables. *Pattern Recognition*, 47(10), 3304–3313.
- Bian, W., Ding, S., & Xue, Y. (2017a). Combining weighted linear project analysis with orientation diffusion for fingerprint orientation field reconstruction. *Information Sciences*, 396, 55–71.
- Bian, W., Ding, S., & Xue, Y. (2017b). Fingerprint image super resolution using sparse representation with ridge pattern prior by classification coupled dictionaries. *IET Biometrics*, 6(5), 342–350.
- Bian, W., Ding, S., & Jia, W. (2018). Collaborative filtering model for enhancing fingerprint image. *IET Image Processing*, 12(1), 149–157.
- Bian, W., Xu, D., Li, Q., Cheng, Y., Jie, B., & Ding, X. (2019). A survey of the methods on fingerprint orientation field estimation. *IEEE Access*, 7, 32644–32663.
- Bigun, J., & Granlund, G. H. (1987). Optimal orientation detection of linear symmetry. In *Proceedings of 1st International Conference on Computer Vision* (pp. 433–438).
- Bolle, R., Serior, A. W., Ratha, N. K., & Pankanti, S. (2002). Fingerprint minutiae: A constructive definition. In *Proceedings of ECCV Workshop on Biometric Authentication* (pp. 58–66).
- Branka, S., Marques, O., & Nešković, A. (2019). *Segmentation and separation of overlapped latent fingerprints: Algorithms, techniques, and datasets*. Briefs in Computer Science. Springer.
- Buades, A., Le, T. M., Morel, J., & Vese, L. A. (2010). Fast cartoon + texture image filters. *IEEE Transactions on Image Processing*, 19(8), 1978–1986.
- Can, X., & Lin, Y. (2009). An adaptive algorithm for smoothing fingerprint orientation fields. In *Proceedings of International Conference on Computational Intelligence and Natural Computing*, Wuhan, China (pp. 70–72).
- Canyellas, N., Cantó, E., Forte, G., & López, M. (2005). Hardware–software codesign of a finger- print identification algorithm. In *Proceedings of 5th International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 683–692).
- Cao, K., & Jain, A. K. (2015). Latent orientation field estimation via convolutional neural network. In *Proceedings of International Conference on Biometrics (ICB)*, Phuket, Thailand.
- Cao, K., Liang, J., & Tian, J. (2012). A div-curl regularization model for fingerprint orientation extraction. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems*, Arlington, VA (pp. 231–236).
- Cao, K., Liu, E., & Jain, A. K. (2014). Segmentation and enhancement of latent fingerprints: A coarse to fine ridge structure dictionary. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(9), 1847–1859.
- Cappelli, R., & Maltoni, D. (2009). On the spatial distribution of fingerprint singularities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(4), 742–748.
- Cappelli, R., Maio, D., & Maltoni, D. (1999). Fingerprint classification based on multi-space KL. In *Proceedings of Workshop on Automatic Identification Advances Technologies* (pp. 117–120).
- Cappelli, R., Maio, D., & Maltoni, D. (2000). Synthetic fingerprint-image generation. In *Proceedings of 15th International Conference on Pattern Recognition* (Vol. 3, pp. 475–478).
- Cappelli, R., Maio, D., & Maltoni, D. (2009). Semi-automatic enhancement of very low quality fin- gerprints. In *Proceedings of International Symposium on Image and Signal Processing and Analysis*, Salzburg (pp. 678–683).
- Cappelli, R., Maltoni, D., & Turroni, F. (2012). Fingerprint enhancement using contextual iterative filtering. In *Proceedings International Conference on Biometrics (ICB)*, New Delhi, India.
- Champod, C., Lennard, C. J., Margot, P., & Stoilovic, M. (2016). *Fingerprints and Other Ridge Skin Impressions* (2nd ed.). CRC Press.

- Chang, J. H., & Fan, K. C. (2001). Fingerprint ridge allocation in direct gray-scale domain. *Pattern Recognition*, 34(10), 1907–1925.
- Chen, T., Jiang, X., & Yau, W. (2004a). Fingerprint image quality analysis. In *Proceedings of International Conference on Image Processing* (pp. 1253–1256).
- Chen, X., Tian, J., Cheng, J., & Yang, X. (2004b). Segmentation of fingerprint images using linear classifier. *EURASIP Journal on Applied Signal Processing*, 2004(4), 480–494.
- Chen, Y., Dass, S. C., & Jain, A. K. (2005). Fingerprint quality indices for predicting authentication performance. In *Proceedings of 5th International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 160–170).
- Chen, J., Chan, F., & Moon, Y. S. (2007). Fingerprint matching with minutiae quality score. In *Proceedings of International Conference on Biometrics* (pp. 663–672).
- Chen, F., Feng, J., Jain, A. K., Zhou, J., & Zhang, J. (2011a). Separating overlapped fingerprints. *IEEE Transactions on Information Forensics and Security*, 6(2), 346–359.
- Chen, H., Pang, L., Liang, J., Liu, E., & Tian, J. (2011b). Fingerprint singular point detection based on multiple-scale orientation entropy. *IEEE Signal Processing Letters*, 18(11), 679–682.
- Chen, C., Feng, J., & Zhou, J. (2016). Multi-scale dictionaries based fingerprint orientation field estimation. In *Proceedings of International Conference on Biometrics* (pp. 1–8).
- Cheng, J., & Tian, J. (2004). Fingerprint enhancement with dyadic scale-space. *Pattern Recognition Letters*, 25(11), 1273–1284.
- Chikkerur, S., & Ratha, N. (2005). Impact of singular point detection on fingerprint matching performance. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 207–212).
- Chikkerur, S., Govindaraju, V., Pankanti, S., Bolle, R., & Ratha, N. (2005). Novel approaches for minutiae verification in fingerprint images. In *Proceedings of Workshops on Application of Computer Vision* (Vol. 1, pp. 111–116).
- Chikkerur, S., Cartwright, A. N., & Govindaraju, V. (2007). Fingerprint enhancement using STFT analysis. *Pattern Recognition*, 40(1), 198–211.
- Chugh, T., Arora, S. S., Jain, A. K., & Paultre, N. G. (2017). Benchmarking fingerprint minutiae extractors. In *Proceedings of International Conference of the Biometrics Special Interest Group*.
- Coetzee, L., & Botha, E. C. (1993). Fingerprint recognition in low quality images. *Pattern Recognition*, 26(10), 1441–1460.
- Da Costa, J. P., Le Pouliquen, F., Germain, C., & Baylou, P. (2001). New operators for optimized orientation estimation. In *Proceedings of International Conference on Image Processing*.
- Darlow, L. N., & Rosman, B. (2017). Fingerprint minutiae extraction using deep learning. In *Proceedings of International Joint Conference on Biometrics*, Denver, CO (pp. 22–30).
- Dass, S. C. (2004). Markov random field models for directional field and singularity extraction in fingerprint images. *IEEE Transactions on Image Processing*, 13(10), 1358–1367.
- Daugman, J. G. (1985). Uncertainty relation for resolution in space, spatial-frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal Optical Society American*, 2, 1160–1169.
- Deerada, C., Phromsuthirak, K., Rungchokanun, A., & Areekul, V. (2020). Progressive focusing algorithm for reliable pose estimation of latent fingerprints. *IEEE Transactions on Information Forensics and Security*, 15, 1232–1247.
- Ding, B., Wang, H., Chen, P., Zhang, Y., Guo, Z., Feng, J., & Liang, R. (2021). Surface and internal fingerprint reconstruction from optical coherence tomography through convolutional neural network. *IEEE Transactions on Information Forensics and Security*, 16, 685–700.
- Donahue, M. L., & Rokhlin, S. I. (1993). On the use of level curves in image analysis. *CVGIP: Image Understanding*, 57(2), 185–203.

- Donida Labati, R., Genovese, A., Muñoz, E., Piuri, V., & Scotti, F. (2018). A novel pore extraction method for heterogeneous fingerprint images using convolutional neural networks. *Pattern Recognition Letters*, 113, 58–66.
- Duda, R. O., Hart, P. E., & Stork, D. G. (2000). *Pattern classification*, 2nd edn. Wiley.
- Erol, A., Halici, U., & Ongun, G. (1999). Feature selective filtering for ridge extraction. In L.C. Jain, U. Halici, I. Hayashi & S.B. Lee (Eds.), *Intelligent biometric techniques in fingerprint & face recognition*. CRC Press.
- Ezeobiejesi, J., & Bhanu, B. (2017). Latent fingerprint image segmentation using deep neural network. In B. Bhanu & A. Kumar (Eds.), *Deep learning for biometrics*. Springer.
- Fan, L., Wang, S., Wang, H., & Guo, T. (2008). Singular points detection based on zero-pole model in fingerprint images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6), 929–940.
- Farina, A., Kovacs-Vajna, Z. M., Leone, A. (1999). Fingerprint minutiae extraction from skeletonized binary images. *Pattern Recognition*, 32(5), 877–889.
- Feng, J., Shi, Y., & Zhou, J. (2012). Robust and efficient algorithms for separating latent overlapped fingerprints. *IEEE Transactions on Information Forensics and Security*, 7(5), 1498–1510.
- Feng, J., Zhou, J., & Jain, A. K. (2013). Orientation field estimation for latent fingerprint enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(4), 925–940.
- Fitz, A. P., & Green, R. J. (1996). Fingerprint classification using hexagonal fast Fourier transform. *Pattern Recognition*, 29(10), 1587–1597.
- Freeman, W. T., & Adelson, E. H. (1991). The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9), 891–906.
- Fronthaler, H., Kollreider, K., & Bigun, J. (2006). Automatic image quality assessment with application in biometrics. In *Proceedings of CVPR Workshop on Biometrcis* (pp. 30–35).
- Fronthaler, H., Kollreider, K., & Bigun, J. (2007). Pyramid-based image enhancement of fingerprints. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 45–50).
- Fronthaler, H., Kollreider, K., & Bigun, J. (2008a). Local features for enhancement and minutiae extraction in fingerprints. *IEEE Transactions on Image Processing*, 17(3), 354–363.
- Fronthaler, H., Kollreider, K., Bigun, J., Fierrez, J., Alonso-Fernandez, F., Ortega-Garcia, J., & Gonzalez-Rodriguez, J. (2008b). Fingerprint image-quality estimation and its application to multialgorithm verification. *IEEE Transactions on Information Forensics and Security*, 3(2), 331–338.
- Gall, J., & Lempitsky, V. (2013). Class-specific Hough forests for object detection. In A. Criminisi & J. Shotton (Eds.), *Decision forests for computer vision and medical image analysis*. Springer.
- Galton, F. (1892). *Finger prints*. Macmillan.
- Gamassi, M., Piuri, V., & Scotti, F. (2005). Fingerprint local analysis for high-performance minutiae extraction. In *Proceedings of International Conference on Image Processing* (Vol. 3, pp. 265–268).
- Girshick, R. (2015). Fast R-CNN. In *Proceedings of International Conference on Computer Vision* (pp. 1440–1448).
- Gonzales, R. C., & Woods, R. E. (2007). *Digital image processing*, 3rd edn. Prentice-Hall.
- Gottschlich, C. (2012). Curved-region-based ridge frequency estimation and curved gabor filters for fingerprint image enhancement. *IEEE Transactions on Image Processing*, 21(4), 2220–2227.
- Gottschlich, C., & Schönliefb, C. (2012). Oriented diffusion filtering for enhancing low-quality fingerprint images. *IET Biometrics*, 1(2), 105–113.
- Gottschlich, C., Mihailescu, P., & Munk, A. (2009). Robust orientation field estimation and extrapolation using semilocal line sensors. *IEEE Transactions on Information Forensics and Security*, 4(4), 802–811.

- Gottschlich, C., Tams, B., & Huckemann, S. (2017). Perfect fingerprint orientation fields by locally adaptive global models. *IET Biometrics*, 6(3), 183–190.
- Govindaraju, V., Shi, Z., & Schneider, J. (2003). Feature extraction using a chaincoded contour representation of fingerprint images. In *Proceedings of 4th International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 268–275.)
- Grasselli, A. (1969). On the automatic classification of fingerprints. In S. Watanabe (Ed.), *Methodologies of pattern recognition*. Academic.
- Greenberg, S., Aladjem, M., Kogan, D., & Dimitrov, I. (2000). Fingerprint image enhancement using filtering techniques. In *Proceedings of 15th International Conference on Pattern Recognition* (Vol. 3, pp. 326–329).
- Grosz, S. A., Engelsma, J. J., Paulte, N. G., & Jain, A. K. (2020). White-box evaluation of fingerprint matchers: Robustness to minutiae perturbations. In *Proceedings of International Joint Conference on Biometrics* (pp. 1–10).
- Grother, P., & Tabassi, E. (2007). Performance of biometric quality measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4), 531–543.
- Grother, P., McCabe, M., Watson, C., Indovina, M., Salamon, W., Flanagan, P., Tabassi, E., Newton, E., & Wilson, C. (2006). *Performance and Interoperability of the INCITS 378 Fingerprint Template*. NIST Research Report: NISTIR 7296.
- Gu, J., Zhou, J., & Yang, C. (2006). Fingerprint recognition by combining global structure and local cues. *IEEE Transactions on Image Processing*, 15(7), 1952–1964.
- Gu, S., Feng, J., Lu, J., & Zhou, J. (2018). Efficient rectification of distorted fingerprints. *IEEE Transactions on Information Forensics and Security*, 13(1), 156–169.
- Hara, M. (2011a). System for recognizing fingerprint image, method and program for the same. US Patent 8019132.
- Hara, M. (2011b). Fingerprint collation apparatus, fingerprint pattern area extracting apparatus and quality judging apparatus, and method and program of the same. US Patent 7885437.
- Hari, V. S., Jagathy Raj, V. P., & Gopikakumari, R. (2013). Unsharp masking using quadratic filter for the enhancement of fingerprints in noisy background. *Pattern Recognition*, 46(12), 3198–3207.
- He, Y., Tian, J., Luo, X., & Zhang, T. (2003). Image enhancement and minutiae matching in fingerprint verification. *Pattern Recognition Letters*, 24(9), 1349–1360.
- Henry, E. (1900). *Classification and uses of finger prints*.
- Hicklin, R. A. (2009). Anatomy of friction ridge skin. In S. Z. Li & A. K. Jain (Eds.), *Encyclopedia of biometrics*. Springer.
- Hong, L., Jain, A. K., Pankanti, S., & Bolle, R. (1996). Fingerprint enhancement. In *Proceedings of Workshop on Applications of Computer Vision* (pp. 202–207).
- Hong, L., Wan, Y., & Jain, A. K. (1998). Fingerprint image enhancement: Algorithms and performance evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8), 777–789.
- Hotz, T. (2009). Intrinsic coordinates for fingerprints based on their longitudinal axis. In *Proceedings of International Symposium on Image and Signal Processing and Analysis*, Salzburg (pp. 500–504).
- Hou, Z., & Yau, W. (2010). A variational formulation for fingerprint orientation modelling. In *Proceedings of International Conference on Pattern Recognition*, Istanbul (pp. 1626–1629).
- Hsieh, C. T., Lai, E., & Wang, Y. C. (2003). An effective algorithm for fingerprint image enhancement based on wavelet transform. *Pattern Recognition*, 36(2), 303–312.
- Huang, C. Y., Liu, L. M., & Hung, D. C. D. (2007). Fingerprint analysis and singular point detection. *Pattern Recognition Letters*, 28(15), 1937–1945.

- Huckemann, S., Hotz, T., & Munk, A. (2008). Global models for the orientation field of fingerprints: An approach based on quadratic differentials. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(9), 1507–1519.
- Hung, D. C. D. (1993). Enhancement and feature purification of fingerprint images. *Pattern Recognition*, 26(11), 1661–1671.
- Hung, D. C. D., & Huang, C. (1996). A model for detecting singular points of a fingerprint. In *Proceedings of 9th Florida Artificial Intelligence Research Symposium* (pp. 444–448).
- Hwang, K. (2004). Statistical quality assessment of a fingerprint. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification I*.
- Ikeda, N., Nakanishi, M., Fujii, K., Hatano, T., Shigematsu, S., Adachi, T., Okazaki, Y., & Kyuragi, H. (2002). Fingerprint image enhancement by pixel-parallel processing. In *Proceedings of 16th International Conference on Pattern Recognition* (Vol. 3, pp. 752–755).
- ISO/IEC 19794-2. (2011). ISO, “ISO/IEC 19794-2:2011 – Information technology – Biometric data interchange formats – Part 2: Finger minutiae data”. Retrieved July, 2021, from <https://www.iso.org/standard/50864.html>.
- ISO/IEC 29794-4. (2017). *Information technology—Biometric sample quality—Part 4: Finger image data*. ISO/IEC Standard.
- Isola, P., Zhu, J., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of Conference on Computer Vision and Pattern Recognition* (pp. 5967–5976).
- Jain, A. K., & Farrokhnia, F. (1991). Unsupervised texture segmentation using Gabor filters. *Pattern Recognition*, 24(12), 1167–1186.
- Jain, A. K., Hong, L., Pankanti, S., & Bolle, R. (1997). An identity authentication system using fingerprints. *Proceedings of the IEEE*, 85(9), 1365–1388.
- Jain, A. K., Prabhakar, S., Hong, L., & Pankanti, S. (2000). Filterbank-based fingerprint matching. *IEEE Transactions on Image Processing*, 9(5), 846–859.
- Jain, A. K., Chen, Y., & Demirkus, M. (2007). Pores and ridges: High-resolution fingerprint matching using Level 3 features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(1), 15–27.
- Jang, W., Park, D., Lee, D., & Kim, S. J. (2006). Fingerprint image enhancement based on a half gabor filter. In *Proceedings of International Conference on Biometrics* (pp. 258–264).
- Jang, H., Kim, D., Mun, S., Choi, S., & Lee, H. (2017). DeepPore: Fingerprint pore extraction using deep convolutional neural networks. *Signal Processing Letters*, 24(12), 1808–1812.
- Ji, L., & Yi, Z. (2008). Fingerprint orientation field estimation using ridge projection. *Pattern Recognition*, 41(5), 1508–1520.
- Ji, L., Yi, Z., Shang, L., & Pu, X. (2007). Binary fingerprint image thinning using template-based PCNNs. *IEEE Transaction on Systems, Man, and Cybernetics, Part B*, 37(5), 1407–1413.
- Jiang, X. (2000). Fingerprint image ridge frequency estimation by higher order spectrum. In *Proceedings of International Conference on Image Processing*.
- Jiang, X. (2001). A study of fingerprint image filtering. In *Proceedings of International Conference on Image Processing*.
- Jiang, X., Yau, W. Y., & Ser, W. (1999). Minutiae extraction by adaptive tracing the gray level ridge of the fingerprint image. In *Proceedings of International Conference on Image Processing*.
- Jiang, X., Yau, W. Y., & Ser, W. (2001). Detecting the fingerprint minutiae by adaptive tracing the gray-level ridge. *Pattern Recognition*, 34(5), 999–1013.
- Jiang, X., Liu, M., & Kot, A. C. (2004). Reference point detection for fingerprint recognition. In *Proceedings of 17th International Conference on Pattern Recognition* (Vol. 1, pp. 540–543).

- Jiang, L., Zhao, T., Bai, C., Yong, A., & Wu, M. (2016). A direct fingerprint minutiae extraction approach based on convolutional neural networks. In *Proceedings of International Joint Conference on Neural Networks*, Vancouver, BC (pp. 571–578).
- Jirachaweng, S., & Areekul, V. (2007). Fingerprint enhancement based on discrete cosine transform. In *Proceedings of International Conference on Biometrics* (pp. 96–105).
- Jirachaweng, S., Hou, Z., Yau, W. Y., & Areekul, V. (2011). Residual orientation modeling for fingerprint enhancement and singular point detection. *Pattern Recognition*, 44(2), 431–442.
- Jolliffe, I. T. (1986). *Principle component analysis*. Springer.
- Joshi, I., Anand, A., Vatsa, M., Singh, R., Roy, S. D., & Kalra, P. (2019). Latent fingerprint enhancement using generative adversarial networks. In *Proceedings Winter Conference on Applications of Computer Vision*, Waikoloa Village, HI, USA (pp. 895–903).
- Kamei, T. (2004). Image filter design for fingerprint enhancement. In N. Ratha & R. Bolle (Eds.), *Automatic fingerprint recognition systems* (pp. 113–126). Springer.
- Kamei, T., & Mizoguchi, M. (1995). Image filter design for fingerprint enhancement. In *Proceedings of International Symposium on Computer Vision* (pp. 109–114).
- Karu, K., & Jain, A. K. (1996). Fingerprint classification. *Pattern Recognition*, 29(3), 389–404.
- Kass, M., & Witkin, A. (1987). Analyzing oriented patterns. *Computer Vision Graphics and Image Processing*, 37(3), 362–385.
- Kawagoe, M., & Tojo, A. (1984). Fingerprint pattern classification. *Pattern Recognition*, 17, 295–303.
- Kayaoglu, M., Topcu, B., & Uludag, U. (2013). Standard fingerprint databases: Manual minutiae labeling and matcher performance analyses. [arXiv:1305.1443](https://arxiv.org/abs/1305.1443).
- Khan, M. A. U., Khan, T. M., Bailey, D. G., & Kong, Y. (2016). A spatial domain scar removal strategy for fingerprint image enhancement. *Pattern Recognition*, 60, 258–274.
- Kim, D. H. (2005). Minutiae quality scoring and filtering using a neighboring ridge structural analysis on a thinned fingerprint image. In *Proceedings of 5th International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 674–682).
- Kim, B. G., & Park, D. J. (2002). Adaptive image normalisation based on block processing for enhancement of fingerprint image. *Electronics Letters*, 38(14), 696–698.
- Kim, S., Lee, D., & Kim, J. (2001). Algorithm for detection and elimination of false minutiae in fingerprint images. In *Proceedings of 3rd International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 235–240).
- Kohonen, T., Kangas, J., Laaksonen, J., & Torkkola, K. (1992). LVQ_PAQ: A program package for the correct application of learning vector quantization algorithms. In *Proceedings of International Joint Conference On Neural Network* (pp. 1725–1730).
- Koo, W. M., & Kot, A. (2001). Curvature-based singular points detection. In *Proceedings of 3rd International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 229–234).
- Kovacs-Vajna, Z. M. (2000). A fingerprint verification system based on triangular matching and dynamic time warping. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22, 1266–1276.
- Kovacs-Vajna, Z. M., Rovatti, R., & Frazzoni, M. (2000). Fingerprint ridge distance computation methodologies. *Pattern Recognition*, 33(1), 69–80.
- Kryszczuk, K., & Drygajlo, A. (2006). Singular point detection in finger-prints using quadrant change information. In *Proceedings of 18th International Conference on Pattern Recognition* (Vol. 4, pp. 594–597).
- Kryszczuk, K. M., Morier, P., & Drygajlo, A. (2004). Study of the distinctiveness of level 2 and level 3 features in fragmentary fingerprint comparison. In *Proceedings of ECCV Workshop on Biometric Authentication* (pp. 124–133).

- Lam, L., Lee, S. W., & Suen, C. Y. (1992). Thinning methodologies: A comprehensive survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(9), 869–885.
- Landy, M. S., Cohen, Y., & Sperling, G. (1984). Hips: A Unix-based image processing system. *Computer Vision, Graphics and Image Processing*, 25(3), 331–347.
- Larkin, K. G. (2005). Uniform estimation of orientation using local and nonlocal 2-D energy operators. *Optics Express*, 13(20), 8097–8121.
- Larkin, K. G., & Fletcher, P. A. (2007). A coherent framework for fingerprint analysis: Are fingerprints holograms? *Optics Express*, 15(14), 8667–8677.
- Le, T. H., & Van, H. T. (2012). Fingerprint reference point detection for image retrieval based on symmetry and variation. *Pattern Recognition*, 45(9), 3360–3372.
- Lee, K., & Prabhakar, S. (2008). Probabilistic orientation field estimation for fingerprint enhancement and verification. In *Proceedings on Biometric Symposium*.
- Lee, B., Moon, J., & Kim, H. (2005). A novel measure of fingerprint image quality using the Fourier spectrum. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification II*.
- Lee, C., Lee, S., Kim, J., & Kim, S. J. (2006). Preprocessing of a fingerprint image captured with a mobile camera. In *Proceedings of International Conference on Biometrics* (pp. 348–355).
- Lehtihet, R., El Oraiby, W., & Benmohammed, M. (2014). Ridge frequency estimation for low-quality fingerprint images enhancement using Delaunay triangulation. *International Journal of Pattern Recognition and Artificial Intelligence*, 28(1), 1456002.
- Leung, M., Engeler, W., & Frank, P. (1990). Fingerprint image processing using neural network. In *Proceedings of IEEE Region 10 Conference on Computer and Communications Systems*.
- Leung, W. F., Leung, S. H., Lau, W. H., & Luk, A. (1991). Fingerprint recognition using neural network. In *Proceedings of Workshop Neural Network for Signal Processing*.
- Li, J., Yau, W. Y., & Wang, H. (2006). Constrained nonlinear models of fingerprint orientations with prediction. *Pattern Recognition*, 39(1), 102–114.
- Li, G., Busch, C., & Yang, B. (2014). A novel approach used for measuring fingerprint orientation of arch fingerprint. In *Proceedings of International Convention on Information and Communication Technology, Electronics and Microelectronics*, Opatija (pp. 1309–1314).
- Li, J., Feng, J., & Kuo, C. C. J. (2018). Deep convolutional neural network for latent fingerprint enhancement. *Signal Processing: Image Communication*, 60, 52–63.
- Liang, X., & Asano, T. (2006). A linear time algorithm for binary fingerprint image denoising using distance transform. *IEICE Transactions on Information and Systems*, 89(4), 1534–1542.
- Lim, E., Jiang, X., & Yau, W. (2002). Fingerprint quality and validity analysis. In *Proceedings of International Conference on Image Processing* (Vol. 1, pp. 469–472).
- Lim, E., Toh, K. A., Suganthan, P. N., Jiang, X., & Yau, W. Y. (2004). Fingerprint image quality analysis. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 1241–1244).
- Lin, W., & Dubes, R. (1983). A review of ridge counting in dermatoglyphics. *Pattern Recognition*, 16(1), 1–8.
- Liu, J., Huang, Z., & Chan, K. (2000). Direct minutiae extraction from gray-level fingerprint image by relationship examination. In *Proceedings of International Conference on Image Processing*.
- Liu, M., Jiang, X., & Kot, A. C. (2004). Fingerprint reference point detection. In *Proceedings of International Conference on Biometric Authentication* (pp. 272–279).
- Liu, T., Zhu, G., Zhang, C., & Hao, P. (2005). Fingerprint indexing based on singular point correlation. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 293–296).
- Liu, T., Zhang, C., & Hao, P. (2006). Fingerprint reference point detection based on local axial symmetry. In *Proceedings of 18th International Conference on Pattern Recognition* (Vol. 1, pp. 1050–1053).

- Liu, S., & Liu, M. (2012). Fingerprint orientation modeling by sparse coding. In *Proceedings of International Conference on Biometrics*, New Delhi (pp. 176–181).
- Liu, M., Liu, S., & Zhao, Q. (2014). Fingerprint orientation field reconstruction by weighted discrete cosine transform. *Information Sciences*, 268, 65–77.
- Liu, E., & Cao, K. (2016). Minutiae extraction from level 1 features of fingerprint. *IEEE Transactions on Information Forensics and Security*, 11(9), 1893–1902.
- Liu, S., Liu, M., & Yang, Z. (2017). Sparse coding based orientation estimation for latent fingerprints. *Pattern Recognition*, 67, 164–176.
- Liu, J., Yan, J., Deng, D., & Zhang, R. (2020). Fingerprint image quality assessment based on BP neural network with hierarchical clustering. *IET Information Security*, 14(2), 185–195.
- Luo, X., & Tian, J. (2000). Knowledge based fingerprint image enhancement. In *Proceedings of 15th International Conference on Pattern Recognition* (Vol. 4, pp. 783–786).
- Ma, C., & Zhu, Y. (2013). Analysis and extraction of fingerprint features based on principal curves. *Journal of Computational Information Systems*, 9(21), 8591–8601.
- Maio, D., & Maltoni, D. (1996). A structural approach to fingerprint classification. In *Proceedings of 13th International Conference on Pattern Recognition*.
- Maio, D., & Maltoni, D. (1997). Direct gray-scale minutiae detection in fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(1).
- Maio, D., & Maltoni, D. (1998a). Ridge-line density estimation in digital images. In *Proceedings of 14th International Conference on Pattern Recognition* (pp. 1654–1658).
- Maio, D., & Maltoni, D. (1998b). Neural network based minutiae filtering in fingerprints. In *Proceedings of 14th International Conference on Pattern Recognition* (pp. 1654–1658).
- Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L., & Jain, A. K. (2002). FVC2000: Fingerprint verification competition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3), 402–412.
- Malathi, S., Uma Maheswari, S., & Meena, C. (2010). Fingerprint pore extraction based on marker controlled watershed segmentation. In *Proceedings of International Conference on Computer and Automation Engineering*, Singapore (pp. 337–340).
- Mallat, S. G. (1989). A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7), 674–693.
- Mehltre, B. M. (1993). Fingerprint image analysis for automatic identification. *Machine Vision and Applications*, 6, 124–139.
- Mehltre, B. M., Murthy, N. N., Kapoor, S., & Chatterjee, B. (1987). Segmentation of fingerprint images using the directional image. *Pattern Recognition*, 20(4), 429–435.
- Mei, Y., Sun, H., & Xia, D. (2009). A gradient-based combined method for the computation of fingerprints' orientation field. *Image and Vision Computing*, 27(8), 1169–1177.
- Miao, D., Tang, Q., & Fu, W. (2007). Fingerprint minutiae extraction based on principal curves. *Pattern Recognition Letters*, 28(16), 2184–2189.
- Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., & Terzopoulos, D. (2021). Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. <https://doi.org/10.1109/TPAMI.2021.3059968>.
- Moayer, B., & Fu, K. (1986). A tree system approach for fingerprint pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(3), 376–388.
- Munir, M. U., Javed, M. Y., & Khan, S. A. (2012). A hierarchical k-means clustering based finger print quality classification. *Neurocomputing*, 85, 62–67.
- Nakamura, T., Hirooka, M., Fujiwara, H., & Sumi, K. (2004). Fingerprint image enhancement using a parallel ridge filter. In *Proceedings of 17th International Conference on Pattern Recognition* (Vol. 1, pp. 536–539).

- Nguyen, D., Cao, K., & Jain, A. K. (2018a). Automatic latent fingerprint segmentation. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems*.
- Nguyen, D., Cao, K., & Jain, A. K. (2018b). Robust minutiae extractor: Integrating deep networks and fingerprint domain knowledge. In *Proceedings of International Conference on Biometrics*.
- Nguyen, V. H., Liu, J., Nguyen, T. H. B., & Kim, H. (2020). Universal fingerprint minutiae extractor using convolutional neural networks. *IET Biometrics*, 9(2), 47–57.
- Nilsson, K., & Bigun, J. (2001). Using linear symmetry features as a pre-processing step for fingerprint images. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 247–252).
- Nilsson, K., & Bigun, J. (2002a). Complex filters applied to fingerprint images detecting prominent points used alignment. In *Proceedings of ECCV Workshop on Biometric Authentication* (pp. 39–47). Springer.
- Nilsson, K., & Bigun, J. (2002b). Prominent symmetry points as landmarks in fingerprint images for alignment. In *Proceedings of 16th International Conference on Pattern Recognition* (Vol. 3, pp. 395–398).
- Nilsson, K., & Bigun, J. (2003). Localization of corresponding points in fingerprints by complex filtering. *Pattern Recognition Letters*, 24(13), 2135–2144.
- NIST. (2015). Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information. Update 2015 of NIST Special Publication 500-290e3.
- Novikov, S. O., & Kot, V. S. (1998). Singular feature detection and classification of fingerprints using hough transform. In *Proceedings of SPIE (6th International Workshop on Digital Image Processing and Computer Graphics: Applications in Humanities and Natural Sciences)* (Vol. 3346, pp. 259–269).
- O’Gorman, L., & Nickerson, J. (1988). Matched filter design for fingerprint image enhancement. In *Proceedings of International Conference on Acoustic Speech and Signal Processing* (pp. 916–919).
- O’Gorman, L., & Nickerson, J. V. (1989). An approach to fingerprint filter design. *Pattern Recognition*, 22(1), 29–38.
- Ohtsuka, T., & Kondo, A. (2005). A new approach to detect core and delta of the fingerprint using extended relational graph. In *Proceedings of International Conference on Image Processing* (Vol. 3, pp. 249–252).
- Ohtsuka, T., & Takahashi T. (2005). A new detection approach for the fingerprint core location using extended relation graph. *IEICE Transactions on Information and Systems*, 88(10), 2308–2312.
- Ohtsuka, T., & Watanabe, D. (2010). Singular candidate method: Improvement of extended relational graph method for reliable detection of fingerprint singularity. *IEICE Transactions on Information and Systems*, E93-D(7), 1788–1797.
- Oliveira, M. A., & Leite, N. J. (2008). A multiscale directional operator and morphological tools for reconnecting broken ridges in fingerprint images. *Pattern Recognition*, 41(1), 367–377.
- Orczyk, T., & Wieclaw, L. (2011). Fingerprint ridges frequency. In *Proceedings World Congress on Nature and Biologically Inspired Computing*, Salamanca (pp. 558–561).
- Ouyang, J., Feng, J., Lu, J., Guo, Z., & Zhou, J. (2017). Fingerprint pose estimation based on faster R-CNN. In *Proceedings of International Joint Conference on Biometrics*, Denver, CO (pp. 268–276).
- Pais Barreto Marques, A. C., & Gay Thome, A. C. (2005). A neural network fingerprint segmentation method. In *Proceedings of International Conference on Hybrid Intelligent Systems*.
- Paiva, A. R. C., & Tasdizen, T. (2012). Fingerprint image segmentation using data manifold characteristic features. *International Journal of Pattern Recognition and Artificial Intelligence*, 26(4), 1256010.

- Panetta, K., Kamath, K. M. S., Rajeev, S., & Agaian, S. S. (2019). LQM: localized quality measure for fingerprint image enhancement. *IEEE Access*, 7, 104567–104576.
- Perona, P. (1998). Orientation diffusions. *IEEE Transactions on Image Processing*, 7(3), 457–467.
- Prabhakar, S., Jain, A. K., & Pankanti, S. (2003). Learning fingerprint minutiae location and type. *Pattern Recognition*, 36(8), 1847–1857.
- Qi, J., Shi, Z., Zhao, X., & Wang, Y. (2005a). Measuring fingerprint image quality using gradient. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification II*.
- Qi, J., Abdurrahim, D., Li, D., & Kunieda, H. (2005b). A hybrid method for fingerprint image quality calculation. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 124–129).
- Ram, S., Bischof, H., & Birchbauer, J. (2010). Modelling fingerprint ridge orientation using Legendre polynomials. *Pattern Recognition*, 43(1), 342–357.
- Rama, R. K. N. V., & Namboodiri, A. M. (2011). Fingerprint enhancement using hierarchical Markov random fields. In *Proceedings of International Joint Conference on Biometrics*, Washington, DC (pp. 1–8).
- Rämö, P., Tico, M., Onnia, V., & Saarinen, J. (2001). Optimized singular point detection algorithm for fingerprint images. In *Proceedings of International Conference on Image Processing*.
- Rao, A. R. (1990). *A taxonomy for texture description and identification*. Springer.
- Ratha, N. K., Chen, S. Y., & Jain, A. K. (1995). Adaptive flow orientation-based feature extraction in fingerprint images. *Pattern Recognition*, 28(11), 1657–1672.
- Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In *Proceedings of Advances in Neural Information Processing Systems* (pp. 91–99).
- Rerkrai, K., & Areekul, V. (2000). A new reference point for fingerprint recognition. In *Proceedings of International Conference on Image Processing*.
- Roddy, A., & Stosz, J. (1997). Fingerprint features: Statistical-analysis and system performance estimates. *Proceedings of the IEEE*, 85(9), 1390–1421.
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of International Conference on Medical Image Computing and Computer Assisted Intervention* (pp. 234–241).
- Saleh, A. M., Bahaa Eldin, A. M., & Wahdan, A. A. (2009). A modified thinning algorithm for fingerprint identification systems. In *Proceedings of International Conference on Computer Engineering & Systems*, Cairo (pp. 371–376).
- Sankaran, A., Vatsa, M., & Singh, R. (2013). Automated clarity and quality assessment for latent fingerprints. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems*, Arlington, VA (pp. 1–6).
- Sankaran, A., Pandey, P., Vatsa, M., & Singh, R. (2014). On latent fingerprint minutiae extraction using stacked denoising sparse AutoEncoders. In *Proceedings International Joint Conference on Biometrics*, Clearwater, FL (pp. 1–7).
- Schuch, P., Schulz, S., & Busch, C. (2016). De-convolutional auto-encoder for enhancement of fingerprint samples. In *Proceedings of International Conference on Image Processing Theory, Tools and Applications*, Oulu (pp. 1–7).
- Schuch, P., Schulz, S., & Busch, C. (2017a). Deep expectation for estimation of fingerprint orientation fields. In *Proceedings of International Joint Conference on Biometrics*, Denver, CO (pp. 185–190).
- Schuch, P., Schulz, S., & Busch, C. (2017b). Intrinsic limitations of fingerprint orientation estimation. In *Proceedings of International Conference of the Biometrics Special Interest Group*, Darmstadt (pp. 1–5).

- Schuch, P., May, J. M., & Busch, C. (2018a). Unsupervised learning of fingerprint rotations. In *Proceedings of International Conference of the Biometrics Special Interest Group*, Darmstadt (pp. 1–6).
- Schuch, P., Schulz, S., & Busch, C. (2018b). Survey on the impact of fingerprint image enhancement. *IET Biometrics*, 7(2), 102–115.
- Sha, L., Zhao, F., & Tang, X. (2006). Minutiae-based fingerprint matching using subset combination. In *Proceedings of International Conference on Pattern Recognition* (Vol. 4, pp. 566–569).
- Sharma, R. P., & Dey, S. (2019). Two-stage quality adaptive fingerprint image enhancement using fuzzy C-means clustering based fingerprint quality analysis. *Image and Vision Computing*, 83–84, 1–16.
- Shen, L., Kot, A., & Koo, W. M. (2001). Quality measures of fingerprint images. In *Proceedings of 3rd International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 266–271).
- Shen, Z., Xu, Y., Li, J., & Lu, G. (2019). Stable pore detection for high-resolution fingerprint based on a CNN detector. In *Proceedings of International Conference on Image Processing*, Taipei, Taiwan (pp. 2581–2585).
- Sherlock, B. G. (2004). Computer enhancement and modeling of fingerprint images. In N. Ratha & R. Bolle (Eds.), *Automatic fingerprint recognition systems* (pp. 87–112). Springer.
- Sherlock, B. G., & Monro, D. M. (1993). A model for interpreting fingerprint topology. *Pattern Recognition*, 26(7), 1047–1055.
- Sherlock, B. G., Monro, D. M., & Millard, K. (1992). Algorithm for enhancing fingerprint images. *Electronics Letters*, 28(18), 1720.
- Sherlock, B. G., Monro, D. M., & Millard, K. (1994). Fingerprint enhancement by directional Fourier filtering. *IEE Proceedings Vision Image and Signal Processing*, 141(2), 87–94.
- Shi, Z., & Govindaraju, V. (2006a). A chaincode based scheme for fingerprint feature extraction. *Pattern Recognition Letters*, 27(5), 462–468.
- Shi, Z., & Govindaraju, V. (2006b). Fingerprint image enhancement based on skin profile approximation. In *Proceedings of 18th International Conference on Pattern Recognition* (Vol. 3, pp. 714–717).
- Shi, Z., Wang, Y., Qi, J., & Xu, K. (2004). A new segmentation algorithm for low quality fingerprint image. In *Proceedings of International Conference on Image and Graphics* (pp. 314–317).
- Shin, J. H., Hwang, H. Y., & Chien, I. L. (2006). Detecting fingerprint minutiae by run length encoding scheme. *Pattern Recognition*, 39(6), 1140–1154.
- Singh, K., Gupta, A., & Kapoor, R. (2015). Fingerprint image super-resolution via ridge orientation-based clustered coupled sparse dictionaries. *Journal of Electronic Imaging*, 24(4), 043015.
- Srinivasan, V. S., & Murthy, N. N. (1992). Detection of singular points in fingerprint images. *Pattern Recognition*, 25(2), 139–153.
- Stock, R. M., & Swonger, C. W. (1969). *Development and Evaluation of a Reader of Fingerprint Minutiae*. Technical Report: XM-2478-X-1:13-17, Cornell Aeronautical Laboratory.
- Stoney, D. A., & Thornton, J. I. (1987). A systematic study of epidermal ridge minutiae. *Journal of Forensic Sciences*, 32(5), 1182–1203.
- Stosz, J. D., & Alyea, L. A. (1994). Automated system for fingerprint authentication using pores and ridge structure. In *Proceedings of SPIE (Automatic Systems for the Identification and Inspection of Humans)* (Vol. 2277, pp. 210–223).
- Su, Y., Feng, J., & Zhou, J. (2016). Fingerprint indexing with pose constraint. *Pattern Recognition*, 54, 1–13.
- Sudiro, S. A., Paindavoine, M., & Kusuma, T. M. (2007). Simple fingerprint minutiae extraction algorithm using crossing number on valley structure. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 41–44).

- Sutthiwichaiporn, P., & Areekul, V. (2013). Adaptive boosted spectral filtering for progressive fingerprint enhancement. *Pattern Recognition*, 46(9), 2465–2486.
- Svoboda, J., Monti, F., & Bronstein, M. M. (2017). Generative convolutional networks for latent fingerprint reconstruction. In *Proceedings of International Joint Conference on Biometrics* (pp. 429–436).
- Székely, E., & Székely, V. (1993). Image recognition problems of fingerprint identification. *Microprocessors and Microsystems*, 17(4), 215–218.
- Tabassi, E., Wilson, C., & Watson, C. (2004). *Fingerprint Image Quality*. NIST Research Report: NISTIR 7151.
- Tabassi, E., Olsen, M. A., Makarov, A., & Busch, C. (2013). *Towards NFIQ II lite—Self-organizing Maps for Fingerprint Image Quality assessment*. NIST Interagency Report 79.
- Tabassi, E., Olsen, M., Bausinger, O., Busch, C., Figlarz, A., Fiumara, G., Henniger, O., Merkle, J., Ruhland, T., Schiel, C., & Schwaiger, M. (2021). NFIQ 2.0—NIST Fingerprint Image Quality. NIST-IR 8382. <https://doi.org/10.6028/NIST.IR.8382>. Accessed July 2021.
- Tamura, H. (1978). A comparison of line thinning algorithms from digital topology viewpoint. In *Proceedings of 4th International Conference on Pattern Recognition* (pp. 715–719).
- Tang, Y., Gao, F., & Feng, J. (2017a). Latent fingerprint minutia extraction using fully convolutional network. In *Proceedings of International Joint Conference on Biometrics*, Denver, CO (pp. 117–123).
- Tang, Y., Gao, F., Feng, J., & Liu, Y. (2017b). FingerNet: An unified deep network for fingerprint minutiae extraction. In *Proceedings of International Joint Conference on Biometrics*, Denver, CO (pp. 108–116).
- Tao, X., Yang, X., Cao, K., Wang, R., Li, P., & Tian, J. (2010). Estimation of fingerprint orientation field by weighted 2D Fourier expansion model. In *Proceedings of International Conference on Pattern Recognition*, Istanbul (pp. 1253–1256).
- Tao, X., Yang, X., Zang, Y., Jia, X., & Tian, J. (2012). A novel measure of fingerprint image quality using Principal Component Analysis (PCA). In *Proceedings of International Conference on Biometrics*, New Delhi (pp. 170–175).
- Tashk, A., Helfroush, M. S., & Muhammadpour, M. (2009). Improvement of fingerprint orientation estimation by a modification of fingerprint orientation model based on 2D Fourier expansion (M-FOMFE). In *Proceedings International Conference on Computer, Control and Communication*, Karachi (pp. 1–6).
- Teixeira, R. F. S., & Leite, N. J. (2013). On adaptive fingerprint pore extraction. In *Proceedings of International Conference on Image Analysis and Recognition* (pp. 72–79).
- Teixeira, R. F. S., & Leite, N. J. (2014). Improving pore extraction in high resolution fingerprint images using spatial analysis. In *Proceedings of International Conference on Image Processing*, Paris (pp. 4962–4966).
- Teixeira, R. F. S., & Leite, N. J. (2017). A new framework for quality assessment of high-resolution fingerprint images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(10), 1905–1917.
- Thai, D. H., & Gottschlich, C. (2016). Global variational method for fingerprint segmentation by three-part decomposition. *IET Biometrics*, 5(2), 120–130.
- Trier, O., & Jain, A. K. (1995). Goal-directed evaluation of binarization methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(12), 1191–1201.
- Turroni, F., Maltoni, D., Cappelli, R., & Maio, D. (2011). Improving fingerprint orientation extraction. *IEEE Transactions on Information Forensics and Security*, 6(3), 1002–1013.
- Uchida, K. (2004). Image-based approach to fingerprint acceptability assessment. In *Proceedings of International Conference on Biometric Authentication* (pp. 294–300).

- Van, T. H., & Le, H. T. (2009a). An efficient algorithm for fingerprint reference-point detection. In *Proceedings of International Conference on Computing and Communication Technologies*, Da Nang (pp. 1–7).
- Van, T. H., & Le, H. T. (2009b). Adaptive noisy fingerprint enhancement based on orientation consistency. In *Proceedings of International Conference on Knowledge and Systems Engineering*, Hanoi (pp. 67–72).
- Verma, M. R., Majumdar, A. K., & Chatterjee, B. (1987). Edge detection in fingerprints. *Pattern Recognition*, 20, 513–523.
- Vernon, D. S. G. (1993). Automatic detection of secondary creases in fingerprints. *Optical Engineering*, 32(10), 2616–2623.
- Viola, P., & Jones, M. J. (2001). Rapid object detection using a boosted cascade of simple features. In *Proceedings of International Conference on Computer Vision and Pattern Recognition* (pp. 511–518).
- Vizcaya, P. R., & Gerhardt, L. A. (1996). A nonlinear orientation model for global description of fingerprints. *Pattern Recognition*, 29(7), 1221–1231.
- Wahab, A., Chin, S. H., & Tan, E. C. (1998). Novel approach to automated fingerprint recognition. *IEE Proceedings Vision Image and Signal Processing*, 145(3), 160–166.
- Wahab, A., Tan, E. C., & Jonatan, A. (2004). Direct gray-scale minutiae extraction. In *Proceedings of International Conference on Biometric Authentication* (pp. 280–286).
- Wang, Y., & Hu, J. (2008). estimate singular point rotation by analytical models. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification V*.
- Wang, Y., & Hu, J. (2011). Global ridge orientation modeling for partial fingerprint identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1), 72–87.
- Wang, L., Suo, H., & Dai, M. (2005). Fingerprint image segmentation based on Gaussian–Hermite moments. In *Proceedings of International Conference on Advanced Data Mining and Applications*.
- Wang, Y., Hu, J., & Phillips, D. (2007a). A fingerprint orientation model based on 2D Fourier expansion (FOMFE) and its application to singular-point detection and fingerprint indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4), 573–585.
- Wang, X., Li, J., & Niu, Y. (2007b). Definition and extraction of stable points from fingerprint images. *Pattern Recognition*, 40(6), 1804–1815.
- Wang, W., Li, J., Huang, F., & Feng, H. (2008). Design and implementation of Log-Gabor filter in fingerprint image enhancement. *Pattern Recognition Letters*, 29(3), 301–308.
- Wang, J., Li, J., & Cao, L. (2011). An improved fast thinning algorithm for fingerprint image and its application. *Journal of Computational Information Systems*, 7(7), 2285–2292.
- Watson, C. I. (1993). NIST special database 14, fingerprint database. U.S. National Institute of Standards and Technology.
- Watson, C. I., & Wilson, C. L. (1992). NIST special database 4, fingerprint database. U.S. National Institute of Standards and Technology.
- Watson, C. I., Candela, G. I., & Grother, P. J. (1994). *Comparison of FFTfingerprint filtering methods for neural network classification*. Technical Report: NIST TR 5493, September 1994.
- Weber, D. M. (1992). A cost effective fingerprint verification algorithm for commercial applications. In *Proceedings of South African Symposium on Communication and Signal Processing*.
- Wegstein, J. H. (1982). An automated fingerprint identification system. U.S. Government Publication, U.S. Department of Commerce, National Bureau of Standards, Washington, DC.
- Weng, D., Yin, Y., & Yang, D. (2011). Singular points detection based on multi-resolution in fingerprint images. *Neurocomputing*, 74(17), 3376–3388.
- Willis, A. J., & Myers, L. (2001). A cost-effective fingerprint recognition system for use with low-quality prints and damaged fingertips. *Pattern Recognition*, 34(2), 255–270.

- Wong, W. J., & Lai, S. H. (2020). Multi-task CNN for restoring corrupted fingerprint images. *Pattern Recognition*, 101, 107203.
- Wu, J. C., & Garris, M. D. (2007). Nonparametric statistical data analysis of fingerprint minutiae exchange with two-finger fusion. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification IV*.
- Wu, C., & Govindaraju, V. (2006). Singularity preserving fingerprint image adaptive filtering. In *Proceedings of International Conference on Image Processing* (pp. 313–316).
- Wu, N., & Zhou, J. (2004). Model based algorithm for singular point detection from fingerprint images. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 885–888).
- Wu, C., Zhou, J., Bian, Z., & Rong, G. (2003). Robust crease detection in fingerprint images. In *Proceedings of Conference on Computer Vision and Pattern Recognition* (Vol. II, pp. 505–510).
- Xiang, M., Wu, X., & Hua, Q. (2009). A fast thinning algorithm for fingerprint image. In *Proceedings of International Conference on Information Science and Engineering*, Nanjing (pp. 1039–1042).
- Xiao, Q., & Raafat, H. (1991). Fingerprint image post-processing: A combined statistical and structural approach. *Pattern Recognition*, 24(10), 985–992.
- Yang, J., Liu, L., Jiang, T., & Fan, Y. (2003). A modified Gabor filter design method for fingerprint image enhancement. *Pattern Recognition Letters*, 24(12), 1805–1817.
- Yang, J., Xiong, N., & Vasilakos, A. V. (2013). Two-stage enhancement scheme for low-quality fingerprint images by learning from the images. *IEEE Transactions on Human-Machine Systems*, 43(2), 235–248.
- Yang, X., Feng, J., & Zhou, J. (2014). Localized dictionaries based orientation field estimation for latent fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(5), 955–969.
- Yao, M. Y. S., Pankanti, S., & Hass, N. (2004). Fingerprint quality assessment. In N. Ratha & R. Bolle (Eds.), *Automatic fingerprint recognition systems* (pp. 55–66). Springer.
- Yao, Z., Le bars, J., Charrier, C., & Rosenberger, C. (2015). Quality assessment of fingerprints with minutiae delaunay triangulation. In *Proceedings of International Conference Information Systems Security and Privacy* (pp. 315–321).
- Yao, Z., Le Bars, J., Charrier, C., & Rosenberger, C. (2018). Comparative study of digital fingerprint quality assessment metrics. In *Proceedings of International Conference on Biometrics*, Gold Coast, QLD (pp. 17–22).
- Yin, Y., Wang, Y., & Yang, X. (2005). Fingerprint image segmentation based on quadric surface model. In *Proceedings of 5th International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 647–655).
- Yoon, S., Cao, K., Liu, E., & Jain, A. K. (2013). LFIQ: latent fingerprint image quality. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems*, Washington, D.C.
- Young, M. R., & Elliott, S. J. (2007). Image quality and performance based on henry classification and finger location. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 51–56).
- Zacharias, G. C., Nair, M. S., & Lal, P. S. (2017). Fingerprint reference point identification based on chain encoded discrete curvature and bending energy. *Pattern Analysis and Applications*, 20(1), 253–267.
- Zhan, X., Sun, Z., Yin, Y., & Chu, Y. (2006). Fingerprint ridge distance estimation: Algorithms and the performance. In *Proceedings of International Conference on Biometrics* (pp. 294–301).
- Zhang, Q., & Yan, H. (2007). Fingerprint orientation field interpolation based on the constrained Delaunay triangulation. *International Journal of Information and Systems Sciences*, 3(3), 438–452.

- Zhang, D., Liu, F., Zhao, Q., Lu, G., & Luo, N. (2011a). Selecting a reference high resolution for fingerprint recognition using minutiae and pores. *IEEE Transactions on Instrumentation and Measurement*, 60(3), 863–871.
- Zhang, H., Miao, D., & Zhong, C. (2011b). Modified principal curves based fingerprint minutiae extraction and pseudo minutiae detection. *International Journal of Pattern Recognition and Artificial Intelligence*, 25(8), 1243–1260.
- Zhang, J., Lai, R., & Kuo, C. J. (2012a). Latent fingerprint segmentation with adaptive total variation model. In *Proceedings of International Conference on Biometrics*, New Delhi (pp. 189–195).
- Zhang, J., Lai, R., & Kuo, C. J. (2012b). Latent fingerprint detection and segmentation with a directional total variation model. In *Proceedings of International Conference on Image Processing* (pp. 1145–1148).
- Zhang, J., Lai, R., & Kuo, C. J. (2013). Adaptive directional total-variation model for latent fingerprint segmentation. *IEEE Transactions on Information Forensics and Security*, 8(8), 1261–1273.
- Zhang, N., Zang, Y., Yang, X., Jia, X., & Tian, J. (2014). Adaptive orientation model fitting for latent overlapped fingerprints separation. *IEEE Transactions on Information Forensics and Security*, 9(10), 1547–1556.
- Zhao, Q., & Jain, A. K. (2010). On the utility of extended fingerprint features: A study on pores. In *Proceedings of CVPR Workshop on Biometrics*, San Francisco.
- Zhao, Q., & Jain, A. K. (2012). Model based separation of overlapping latent fingerprints. *IEEE Transactions on Information Forensics and Security*, 7(3), 904–918.
- Zhao, F., & Tang, X. (2007). Preprocessing and postprocessing for skeleton-based fingerprint minutiae extraction. *Pattern Recognition*, 40(4), 1270–1281.
- Zhao, Q., Zhang, L., Zhang, D., Huang, W., & Bai, J. (2009). Curvature and singularity driven diffusion for oriented pattern enhancement with singular points. In *Proceedings of Conference on Computer Vision and Pattern Recognition*, Miami, FL (pp. 2129–2135).
- Zhao, Q., Zhang, D., Zhang, L., & Luo, N. (2010a). High resolution partial fingerprint alignment using pore-valley descriptors. *Pattern Recognition*, 43(3), 1050–1061.
- Zhao, Q., Zhang, D., Zhang, L., & Luo, N. (2010b). Adaptive fingerprint pore modeling and extraction. *Pattern Recognition*, 43(8), 2833–2844.
- Zhao, Q., Liu, F., Zhang, L., & Zhang, D. (2010c). A comparative study on quality assessment of high resolution fingerprint images. In *Proceedings of International Conference on Image Processing*, Hong Kong (pp. 3089–3092).
- Zhou, J., & Gu, J. (2004a). A model-based method for the computation of fingerprints' orientation field. *IEEE Transactions on Image Processing*, 13(6), 821–835.
- Zhou, J., & Gu, J. (2004b). Modeling orientation fields of fingerprints with rational complex functions. *Pattern Recognition*, 37(2), 389–391.
- Zhou, J., Wu, C., Bian, Z., & Zhang, D. (2004). Improving fingerprint recognition based on crease detection. In *Proceedings of International Conference on Biometric Authentication* (pp. 287–293).
- Zhou, J., Gu, J., & Zhang, D. (2007). Singular points analysis in fingerprints based on topological structure and orientation field. In *Proceedings of International Conference on Biometrics* (pp. 261–270).
- Zhou, J., Chen, F., Gu, J. (2009a). A novel algorithm for detecting singular points from fingerprint images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(7), 1239–1250.
- Zhou, J., Chen, F., Wu, N., & Wu, C. (2009b). Crease detection from fingerprint images and its applications in elderly people. *Pattern Recognition*, 42(5), 896–906.
- Zhu, E., Yin, J., & Zhang, G. (2004). Fingerprint enhancement using circular Gabor filter. In *Proceedings of International Conference on Image on Analysis and Recognition* (pp. 750–758).
- Zhu, E., Yin, J., Hu, C., & Zhang, G. (2005). Quality estimation of fingerprint image based on neural network. In *Proceedings of International Conference on Natural Computation* (pp. 65–70).

- Zhu, E., Yin, J., Hu, C., & Zhang, G. (2006). A systematic method for fingerprint ridge orientation estimation and image segmentation. *Pattern Recognition*, 39(8), 1452–1472.
- Zhu, E., Guo, X., & Yin, J. (2016). Walking to singular points of fingerprints. *Pattern Recognition*, 56, 116–128.
- Zhu, Y., Yin, X., Jia, X., & Hu, J. (2017). Latent fingerprint segmentation based on convolutional neural networks. In *Proceedings of Workshop on Information Forensics and Security*, Rennes (pp. 1–6).



Fingerprint Matching

4

Abstract

This chapter formalizes the fingerprint matching problem, namely finding a similarity between any given fingerprint pair. The requirements of the matching problem are high similarity values between two impressions of the same finger and low similarity values between two impressions of different fingers. The intrinsic difficulties in computing this similarity are explained. The three main families of matching approaches are discussed, namely correlation, minutiae, and feature-based methods. Among various approaches available in the literature, the most effective solutions are presented and categorized. In particular, the chapter focuses on the evolution of minutiae matching: from early (global) methods to rich local minutiae descriptors to Minutiae Cylinder Code (MCC). Development of feature-based matching from FingerCode to handcrafted textural features to learning-based deep features is explained. Other important topics such as dense fingerprint registration, distortion correction and pore matching are reviewed. Finally, an overview of benchmarks and evaluation protocols is provided.

Keywords

Fingerprint matching • Similarity • Correlation-based matching • Minutiae-based matching • Feature-based matching • Dense registration • Distortion correction • Global matching • Local descriptors • Learning-based matching • Benchmarks • Databases

Electronic supplementary material The online version contains supplementary material available at ([10.1007/978-3-030-83624-5_4](https://doi.org/10.1007/978-3-030-83624-5_4)).

4.1 Introduction

A fingerprint matching algorithm compares two given fingerprint images and returns either a degree of similarity (without loss of generality, a score between 0 and 1 with 1 indicating maximum similarity) or a binary decision (mated/non-mated). Only a few matching algorithms operate directly on grayscale fingerprint images; most of them require that an intermediate fingerprint representation be derived through a feature extraction stage (refer to Chap. 3). Without loss of generality, hereafter we denote the representation of the fingerprint acquired during enrollment as the *template* (\mathbf{T}) and the representation of the fingerprint to be compared as the *input* (\mathbf{I}). In case no feature extraction is performed, the fingerprint representation coincides with the grayscale fingerprint image itself; hence, throughout this chapter, we denote both raw fingerprint images and fingerprint feature vectors (e.g., minutiae) with \mathbf{T} and \mathbf{I} .

The fingerprint feature extraction and matching algorithms are usually quite similar for both fingerprint verification and identification problems. This is because the fingerprint identification problem (i.e., searching for an input fingerprint in a database of N fingerprints) can be implemented as a sequential execution of N one-to-one comparisons (verifications) between pairs of fingerprints. Fingerprint classification and indexing techniques are usually exploited to speed up the search (refer to Chap. 5) in fingerprint identification problems.

Matching fingerprint images is a difficult problem, mainly due to the large variability in different impressions of the same finger (i.e., large *intra-class* variations). The main factors responsible for intra-class variations are summarized below.

- *Displacement*: the same finger may be placed at different locations on a touch sensor during different acquisitions resulting in a (global) translation of the fingerprint area. A finger displacement of just 2 mm (imperceptible to the user) results in a translation of about 40 pixels in a fingerprint image scanned at a resolution of 500 dpi.
- *Rotation*: the same finger may be rotated at different angles with respect to the sensor surface during different acquisitions. In spite of the finger “guide” mounted in certain commercial scanners, involuntary finger rotations of up to $\pm 20^\circ$ with respect to vertical orientation can be observed in practice.
- *Partial overlap*: finger displacement and rotation often cause part of the fingerprint area to fall outside the sensor’s “field of view,” resulting in a smaller overlap between the foreground areas of the template and the input fingerprints. This problem is particularly serious for small-area touch sensors (see Sect. 2.9) and latent fingerprints.
- *Non-linear distortion*: the act of fingerprint sensing maps the three-dimensional shape of a finger onto the two-dimensional surface of the sensor. This mapping results in a non-linear distortion in successive acquisitions of the same finger due to skin plasticity. Often, fingerprint matching algorithms disregard the characteristic of such a mapping and consider a fingerprint image as non-distorted by assuming that it was produced

by a correct finger placement; a finger placement is correct when: (i) the trajectory of the finger approaching the sensor is orthogonal to the sensor surface; (ii) once the finger touches the sensor surface, the user does not apply traction or torsion. However, due to skin plasticity, the components of the force that are non-orthogonal to the sensor surface produce non-linear distortions (compression or stretching) in the acquired fingerprints. Distortion results in the inability to match fingerprints as rigid patterns.

- *Pressure and skin condition*: the ridge structure of a finger would be accurately captured if ridges of the part of the finger being imaged were in uniform contact with the sensor surface. However, finger pressure, dryness of the skin, skin disease, sweat, dirt, grease, and humidity in the air all confound the situation, resulting in a non-uniform contact. As a consequence, the acquired fingerprint images are typically very noisy and the noise varies in successive acquisitions of the same finger depending on the magnitude of the above cited causes.
- *Noise*: it is mainly introduced by the fingerprint sensing system; for example, residues are left over on the glass platen from the prior fingerprint capture.
- *Feature extraction errors*: the feature extraction algorithms are imperfect and often introduce measurement errors. Errors may be made during any of the feature extraction stages (e.g., estimation of orientation and frequency images, detection of the number, type, and position of the singularities, segmentation of the fingerprint area from the background, etc.). Some enhancement algorithms may perturb the true location and orientation of the minutiae from their gray-scale counterparts. In low-quality fingerprint images, the minutiae extraction process may introduce a large number of spurious minutiae and may not be able to detect all the true minutiae.

The pairs of images in Fig. 4.1 visually show the high variability (large *intra-class* variations) in two different impressions of the same finger.

On the other hand, as evident from Fig. 4.2, fingerprint images from different fingers may sometimes appear quite similar (small *inter-class* variations), especially in terms of global structure (position of the singularity, local ridge orientation, etc.). Although the probability that a large number of minutiae from impressions of two different fingers will match is extremely small (refer to Chap. 8), fingerprint matchers aim to find the “best” alignment. They often tend to declare that a pair of the minutiae “match” even when they are not perfectly coincident.

A large number of automated fingerprint matching algorithms have been proposed in the literature. Most of these algorithms have no difficulty in matching good quality fingerprint images. However, fingerprint matching remains a challenging problem due to the difficulty in matching low-quality and partial (i.e., latent) fingerprints. In the case of human-assisted AFIS, a quality-checking algorithm can be used to ensure that only good quality fingerprints are in the enrollment database. Furthermore, the processing of “difficult” latent fingerprints can be supervised. However, human intervention is not feasible in



Fig. 4.1 Each row shows a pair of impressions of the same finger, taken from the FVC2002 DB1 (Maio et al., 2002b), which were falsely non-matched by most of the matching algorithms submitted to FVC2002. The main cause of difficulty is a very small overlap in the first row, high non-linear distortion in the second row, and very different skin conditions (i.e., dry skin) in the third row



Fig. 4.2 Each row shows a pair of impressions of two different fingers, taken from the FVC2002 databases (Maio et al., 2002b) which were falsely matched by some of the algorithms submitted to FVC2002

unattended on-line fingerprint recognition systems, which are being increasingly deployed in commercial applications (e.g., access control).

An analysis of the false non-match errors produced by the various fingerprint matching algorithms that participated in FVC2000 showed that most of the errors were made on about 20% poor quality fingerprints that constituted the database. In other words, typically, 20% of the database is responsible for about 80% of the false non-match errors (Maio et al., 2002b). Advances in state-of-the-art of fingerprint recognition technology were perceived throughout the different editions of the Fingerprint Verification Competition (BioLab, 2007) and the still running FVC-onGoing.¹ Although a direct comparison across the different competitions is not possible due to the use of databases of unequal difficulty, the accuracy of the top algorithms on FVC-onGoing is an order of magnitude better than the first competition in 2000 (see Sect. 4.7.2):

¹ <https://biolab.csr.unibo.it/fvcongoing>.

- on FV-STD-1.0 database, which was collected under realistic operating conditions with a large area sensor, EER $\cong 0.01\%$;
- on FV-HARD-1.0 database, which includes many challenging images, EER $\cong 0.2\%$.

However, there is still a need to continually develop more robust systems capable of properly processing and comparing poor quality fingerprint images; this is particularly important when dealing with large scale applications (e.g., civil registration or national ID programs), population contains individuals who do manual labor with hands or when small area and low cost sensors are employed, like in the current generation of smartphones. Latent fingerprints deserve special attention during both feature extraction and matching stages: some techniques introduced in this Chapter (and in Chap. 3) are applicable to latent fingerprints as well, Chap. 6 focuses on specific methods introduced to improve latent fingerprint recognition.

Approaches to fingerprint matching can be broadly classified into three families.

- *Correlation-based matching*: two fingerprint images are superimposed, and the correlation between the corresponding pixels is computed for different alignments (e.g., various displacements and rotations). Correlation-based techniques are described in Sect. 4.2.
- *Minutiae-based matching*: this is the most popular and still widely used technique, being the basis of the fingerprint comparison made by fingerprint examiners. Minutiae are extracted from the two fingerprints and stored as sets of two-dimensional points. Minutiae-based matching essentially consists of finding the alignment between the template and the input minutiae feature sets that result in the maximum similarity according to a given score definition (e.g., maximum number of minutiae pairings). Sections 4.3 and 4.4 are dedicated to minutiae matching techniques.
- *Feature-based matching*: the approaches belonging to this family compare fingerprints in term of features extracted from the fingerprint pattern. These features can have a natural counterpart (e.g., pores), can be handcrafted (e.g., SIFT), or learned end-to-end through a deep learning model. In principle, correlation-based and minutiae-based matching could be conceived of as subfamilies of feature-based matching, inasmuch as the pixel intensity and minutiae are themselves features of the finger pattern; throughout this book we address them separately, and in Sect. 4.6 we focus only those matching techniques that use neither minutiae nor pixel intensity as their main fingerprint representation.

As discussed in Sect. 2.9 fingerprint recognition using small area sensors is more complex (and less reliable) due to the limited amount of information in the acquired fragments. Fingerprint mosaicking allows to consolidate several fragments into a full enrolled impression thus enabling a partial-to-full matching at verification time. However, mosaicking is prone to errors and can introduce artefacts at the fragment junctions; hence, some

researchers prefer to leave the enrollment fragments isolated and implement a multiple partial-to-partial fingerprint image comparison. For both the cases, correlation-, minutiae- or feature-based approaches can be used; however, correlation- and feature-based matchings are often preferred for mobile fingerprint recognition because the limited number of minutiae in a fragment might lead to unreliable decisions. More details will be provided in the following Sections. Actually, in an effort to increase the screen size, most mobile phone vendors have gotten rid of the bezel around the screen. This has eliminated the home button, and the small area fingerprint reader. Modern mobile phones either have a face unlock or fingerprint unlock where the optical reader is under the screen.

While most of the literature on fingerprint recognition focuses on software approaches, some researchers proposed hardware-aware implementations. These include early attempts to improve efficiency when computing power was much more limited than today (Gowrisankar, 1989; Ratha et al., 1996a; Prabhakar & Rao, 1989) and recent efforts to speed-up large scale identification on parallel architectures and/or GPU (Cappelli et al., 2015; Ghafoor et al., 2018; Gutiérrez et al., 2014; Lastra et al., 2015; Peralta et al., 2014, 2017). Thanks to both hardware and software improvements, fingerprint identification efficiency has dramatically increased in the last decade and today a single workstation with a GPU can (brute-force) match more than 100 million fingerprints per second (see Sect. 4.4.3).

Finally, to provide a more complete panorama of the techniques proposed in the past, we cite some early studies presented in AFIS, academic, and commercial environments: Banner and Stock (1974, 1975a, b), Millard (1975, 1983), Singh et al. (1977), Hoshino et al. (1980), Liu et al. (1982), Li and Zhang (1984) and Sparrow and Sparrow (1985a, b).

4.2 Correlation-Based Techniques

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

$$SSD(\mathbf{T}, \mathbf{I}) = \|\mathbf{T} - \mathbf{I}\|^2 = (\mathbf{T} - \mathbf{I})^T(\mathbf{T} - \mathbf{I}) = \|\mathbf{T}\|^2 + \|\mathbf{I}\|^2 - 2\mathbf{T}^T\mathbf{I}, \quad (4.1)$$

where the superscript “T” denotes the transpose of a vector and $\| . \|$ denotes the L2 norm of a vector. If the terms $\|\mathbf{T}\|^2$ and $\|\mathbf{I}\|^2$ are constant, the diversity between the two images is minimized when the cross-correlation (*CC*) between \mathbf{T} and \mathbf{I} is maximized:

$$CC(\mathbf{T}, \mathbf{I}) = \mathbf{T}^T\mathbf{I} \quad (4.2)$$

Note that the quantity $-2 \times CC(\mathbf{T}, \mathbf{I})$ appears as the third term in Eq. (4.1). The cross-correlation (or simply correlation) is then a measure of the image similarity. Due to the displacement and rotation that unavoidably characterize two impressions of a given finger, their similarity cannot be simply computed by superimposing \mathbf{T} and \mathbf{I} and applying Eq. (4.2).

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

$$S(\mathbf{T}, \mathbf{I}) = \max_{\Delta x, \Delta y, \theta} CC(\mathbf{T}, \mathbf{I}^{(\Delta x, \Delta y, \theta)}) \quad (4.3)$$

A direct application of Eq. (4.3) rarely leads to acceptable matching results (see Fig. 4.3a) mainly due to the following problems.

1. Non-linear distortion makes impressions of the same finger significantly different in terms of global structure; in particular, the elastic distortion does not significantly alter the fingerprint pattern locally, but since the effects of distortion get integrated in image space, two global fingerprint patterns cannot be reliably correlated (see Fig. 4.3b).
2. Skin condition and finger pressure cause image brightness, contrast, and ridge thickness to vary significantly across different impressions (see Fig. 4.3c). The use of more sophisticated correlation measures such as the *normalized cross-correlation* or the *zero-mean normalized cross-correlation* (Crouzil et al., 1996) may compensate for contrast and brightness variations and applying a proper combination of enhancement, binarization, and thinning steps (performed on both \mathbf{T} and \mathbf{I}) may limit the ridge thickness problem (Kobayashi, 1992). Hatano et al. (2002) proposed using the *differential correlation*, which is computed as the maximum correlation minus the minimum correlation, in a neighborhood of the point where the correlation is maximum. In fact, due to the cyclic nature of fingerprint patterns, if two corresponding portions of the same fingerprint are slightly misaligned with respect to their optimum matching position, the correlation value falls sharply whereas two non-corresponding portions exhibit a flatter correlation value in the neighborhood of the optimum matching position. Hatano et al. (2002) reported a significant accuracy improvement with respect to the conventional correlation method. A very similar technique is known as Peak-to-Sidelobe-Ratio (PSR); see Venkataramani et al. (2005).
3. A direct application of Eq. (4.3) is computationally very expensive. For example, consider two 400×400 pixel images; then the computation of the cross-correlation (Eq. 4.2) for a single value of the $(\Delta x, \Delta y, \theta)$ triplet would require 160,000 multiplications and 160,000 summations (neglecting border effects). If Δx , Δy were both sampled with a one-pixel step in the range $[-200, 200]$ and θ with a step size of 1° in the range $[-30^\circ, 30^\circ]$ we would have to compute $401 \times 401 \times 61$ cross-correlations,

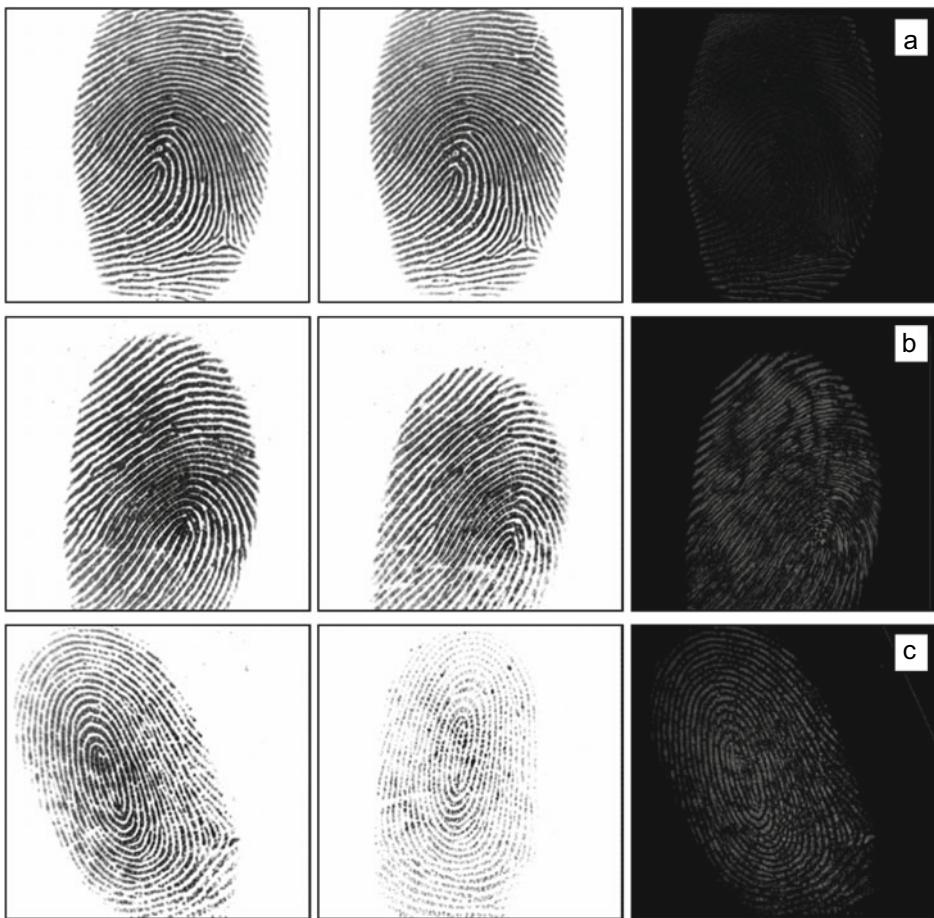


Fig. 4.3 Each row shows two impressions of the same finger and the absolute value of their difference (residual) for the best alignment (i.e., that maximizes correlation). In the first row, **a**, the two impressions are very similar and the corresponding images correlate well (the residual is very small). In the second row, **b**, and third row, **c**, due to high distortion and skin condition, respectively, the residuals are high and the global correlation methods fail.

resulting in about 1.5 trillion multiplications and summations (i.e., about 16 s on a 100,000 MIPS computer).

The fingerprint distortion problem (point 1 in the above list) is usually addressed by computing the correlation locally instead of globally: a set of local regions (whose typical size may be 32×32) is extracted from the template image \mathbf{T} and each of them is independently correlated with the whole input image \mathbf{I} (Bazen et al., 2000). The local regions

may be defined in several ways: (i) their union completely covers \mathbf{T} and their intersection is null (full coverage without any overlap); (ii) their union completely covers \mathbf{T} and they locally overlap (full coverage with overlap); (iii) only certain “interesting” regions are selected from \mathbf{T} . For example, Yahagi et al. (1990), Kovacs-Vajna (2000), and Belez-nai et al. (2001) select small windows around the minutiae, whereas Bazen et al. (2000) consider selective regions that are distinctively localized in the input image (i.e., which fit well at the right location, but do not fit at other locations). It is worth noting that a local correlation approach enables a direct implementation of the partial-to-full matching used with small sensors in mobile devices; in this case the local region coincides with the verification fragment, while the full image corresponds to the mosaic built at enrollment time.

When fingerprint correlation is carried out locally, the correlation estimates in different regions may be simply combined to obtain a similarity measure (e.g., the number of estimates exceeding a certain threshold divided by the total number of estimates). In addition to the values of the correlation, the coordinates of the points where each region has maximum correlation can be exploited to strengthen the matching (*consolidation* or *global optimization* step): in fact, the spatial relationship (distances, angles, etc.) between the regions in the template and their mates in the input image is required to be preserved (Bazen et al., 2000). In the dense fingerprint registration approach by Si et al. (2017), multiple registrations are retained for each region and later consolidated with registrations of neighboring regions through a global optimization (see Fig. 4.4).

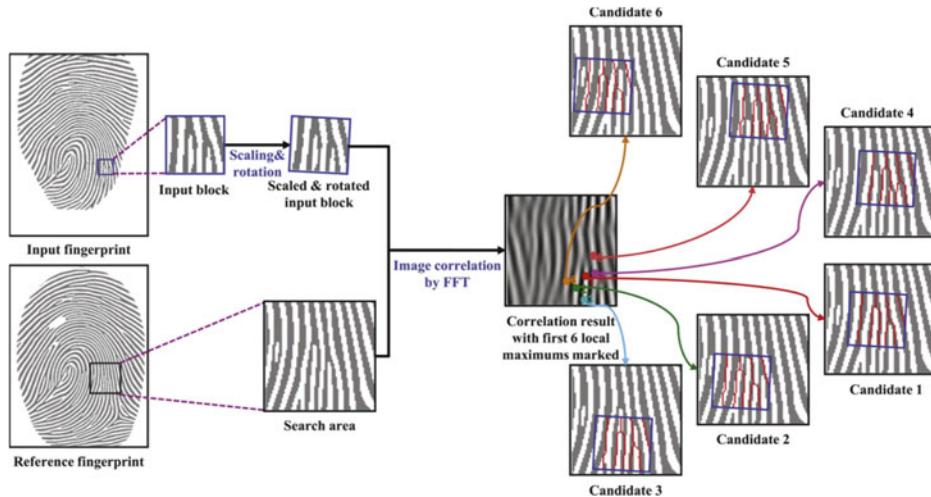


Fig. 4.4 Multiple registrations of a local region in the approach by Si et al. (2017). The 6 registrations corresponding to the highest correlation maximums are shown. © Elsevier. Reprinted, with permission, from Si et al. (2017)

Efficient implementations of the correlation technique have been presented.

- The correlation theorem (Gonzales & Woods, 2007) states that computing the correlation in the spatial domain (operator \otimes) is equivalent to performing a point-wise multiplication in the Fourier domain; in particular,

$$\mathbf{T} \otimes \mathbf{I} = F^{-1}(F^*(\mathbf{T}) \times F(\mathbf{I})) \quad (4.4)$$

where $F()$ is the Fourier transform of an image, $F^{-1}()$ is the inverse Fourier transform, “ $*$ ” denotes the complex conjugate, and “ \times ” denotes the point-by-point multiplication of two vectors. The result of Eq. (4.4) is a correlation image whose value at the pixel $[x, y]$ denotes the correlation between \mathbf{T} and \mathbf{I} when the displacement is $\Delta x = x$ and $\Delta y = y$. However, the output of Eq. (4.4) is dependent on the image energy and the correlation peak (corresponding to the optimal registration) can be small. The Symmetric Phase Only Filter (SPOF) often provides better results (Eq. 4.5):

$$\mathbf{T} \otimes_{\text{SPOF}} \mathbf{I} = F^{-1}\left(\frac{F^*(\mathbf{T})}{|F(\mathbf{T})|} \times \frac{F(\mathbf{I})}{|F(\mathbf{I})|}\right) \quad (4.5)$$

To reduce the effect of noise, Ito et al. (2005, 2006) and Shuai et al. (2007) suggest restricting the SPOF domain to the frequency range characterizing a fingerprint image: this can be simply dealt with through band-pass filtering in the Fourier space. Equations (4.4) and (4.5) do not take into account rotation, which is dealt with separately; in any case, the computational saving is very high when correlation is performed globally (Coetze & Botha, 1993) and considerable when it is performed locally by using medium-size regions. Shabrina et al. (2016) describe a partial-to-full mobile fingerprint recognition approach based on phase-only correlation.

- Computing the maximum correlation need not necessarily be done in a sequential, exhaustive manner; multi-resolution approaches, space-searching techniques (e.g., gradient descent), and other heuristics can be adopted to reduce the number of evaluations. For example, Lindoso et al. (2007) propose to coarsely pre-align the two fingerprints based on their orientation images while Li et al. (2014b) suggest an iterative optimization to find the optimal transformation starting from an initial alignment.
- The Fourier-Mellin transform (Sujan & Mulqueen, 2002) may be used instead of the Fourier transform to achieve rotation invariance in addition to translation invariance; on the other hand, some additional steps (such as the logpolar coordinate transformation) have to be performed, that can reduce the accuracy of this solution. Ouyang et al. (2006) method computes the Fourier-Mellin descriptors locally and uses SPOF to determine the similarity between any two image portions.

- The approach proposed by Wilson et al. (1997) partitions both \mathbf{T} and \mathbf{I} into local regions and computes the maximum correlation (in the Fourier domain) between any pair of regions. This method suffers from “border effects” because of the partial overlap between neighboring blocks, but can considerably speed up the whole matching process.

More sophisticated correlation techniques make use of advanced correlation filters (Kumar et al., 2004), where the template \mathbf{T} is not just a single image but is obtained as a weighted linear combination of several images of the same finger; the weights are determined through an optimization approach (e.g., maximize the correlation peak for fingerprints from the same finger and minimize the correlation peak for fingerprints from other fingers). A preliminary work in this direction was proposed by He et al. (1993). Roberge et al. (1999), Venkataramani and Kumar (2003, 2004), and Watson and Casasent (2004a, b) tested different types of advanced correlation filters for matching distorted fingerprints: the results of their experiments show that these techniques, given a sufficient number of pre-aligned training images, are quite robust and can effectively work with image resolutions lower than 500 dpi.

Correlation-based techniques share characteristics with dense fingerprint registration approaches discussed in Sect. 4.5.4. In fact, the aim of dense image registration is to build an accurate pixel-level correspondence between two fingerprint images of the same finger. However, effective dense registration techniques can be computationally demanding and usually require an initial coarse alignment.

Finally, it is well known that correlation between two images can be computed by an optical system to derive the Fourier transform of the images and a joint transform correlator for their matching. Several systems have been proposed in the literature for optical fingerprint matching: McMahon et al. (1975), Fielding et al. (1991), Grycewicz (1995, 1966, 1999), Rodolfo et al. (1995), Grycewicz and Javidi (1996), Petillot et al. (1996), Soifer et al. (1996), Gamble et al. (1992), Wilson et al. (1997), Kobayashi and Toyoda (1999), Lal et al. (1999), Stoianov et al. (1999), Watson et al. (2000) and Bal et al. (2005) where local minutiae structures are used as input instead of the gray scale image. However, optical systems require complex and expensive hardware/optical components and therefore are not well suited for practical applications.

4.3 Minutiae-Based Methods

Minutiae matching is certainly the most well-known and most widely used method for fingerprint matching, thanks to its strict analogy with the way forensic experts manually compare fingerprints and its acceptance as a proof of identity in the courts of law in almost all countries around the world. This Section focuses on *global* minutiae matching, that is trying to find the best pairing for all the minutiae in the two fingerprints under comparison.

Several approaches exist to reliably solve global minutiae matching; however, their two main weaknesses are the low robustness against distortion and the high computational complexity. The *local* minutiae matching techniques introduced in Sect. 4.4 can overcome these limitations. Note that local minutiae matching can be combined with global minutiae matching by using it in the consolidation step (Sect. 4.4.4).

4.3.1 Problem Formulation

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

$$\begin{aligned}\mathbf{T} &= \{\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_m\}, \mathbf{m}_i = \{x_i, y_i, \theta_i\}, i = 1\dots m \\ \mathbf{I} &= \{\mathbf{m}'_1, \mathbf{m}'_2, \dots, \mathbf{m}'_n\}, \mathbf{m}'_j = \{x'_j, y'_j, \theta'_j\}, j = 1\dots n,\end{aligned}$$

where m and n denote the number of minutiae in \mathbf{T} and \mathbf{I} , respectively.

A minutia \mathbf{m}'_j in \mathbf{I} and a minutia \mathbf{m}_i in \mathbf{T} are considered *paired* or *mated* if the *spatial distance* (sd) between them is smaller than a given tolerance r_0 and the *angular difference* (ad) between them is smaller than an angular tolerance θ_0 :

$$sd(\mathbf{m}'_j, \mathbf{m}_i) = \sqrt{\left(x'_j - x_i\right)^2 + \left(y'_j - y_i\right)^2} \leq r_0, \quad (4.6)$$

$$ad(\mathbf{m}'_j, \mathbf{m}_i) = \min\left(\left|\theta'_j - \theta_i\right|, 360^\circ - \left|\theta'_j - \theta_i\right|\right) \leq \theta_0 \quad (4.7)$$

Equation (4.7) takes the minimum of $\left|\theta'_j - \theta_i\right|$ and $360^\circ - \left|\theta'_j - \theta_i\right|$ because of the circularity of angles (the difference between angles of 2° and 358° is only 4°). The *tolerance boxes* (or hyper-spheres) defined by r_0 and θ_0 are necessary to compensate for the unavoidable errors made by feature extraction algorithms and to account for the small plastic distortions that cause the minutiae positions to change.

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

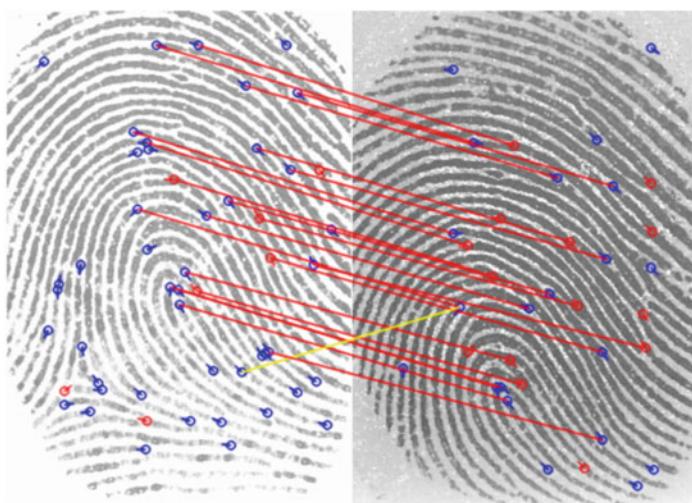


Fig. 4.5 The minutiae pairs resulting from the alignment of two images of the same finger are made explicit by (red) graphical links. The yellow link denotes a false pair

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

In any case, tolerating more geometric transformations beyond translation and rotation results in additional degrees of freedom to the minutiae matcher: when a matcher is designed, this issue needs to be carefully evaluated, as each degree of freedom results in a huge number of new possible alignments which significantly increases the chance of incorrectly matching two fingerprints from different fingers. Figure 4.5 shows coupling links between paired minutiae resulting from the optimal alignment of two fingerprint images.

Let $map()$ be the function that maps a minutia \mathbf{m}'_j (from \mathbf{I}) into \mathbf{m}''_j according to a given geometrical transformation; for example, by considering a displacement of $[\Delta x, \Delta y]$ and a counterclockwise rotation θ around the origin²:

$$map_{\Delta x, \Delta y, \theta}(\mathbf{m}'_j = \{x'_j, y'_j, \theta'_j\}) = \mathbf{m}''_j = \{x''_j, y''_j, \theta'_j + \theta\}$$

² The origin is usually selected as the minutiae centroid (i.e., the average point); before the matching step, minutiae coordinates are adjusted by subtracting the centroid coordinates.

where

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

Let $mm()$ be an indicator function that returns 1 in the case where the minutiae \mathbf{m}''_j and \mathbf{m}_i can be paired according to Eqs. (4.6) and (4.7):

$$mm(\mathbf{m}''_j, \mathbf{m}_i) = \begin{cases} 1 & sd(\mathbf{m}''_j, \mathbf{m}_i) \leq r_0 \text{ and } dd(\mathbf{m}''_j, \mathbf{m}_i) \leq \theta_0 \\ 0 & \text{otherwise.} \end{cases}$$

Then, the matching problem can be formulated as

$$\underset{\Delta x, \Delta y, \theta, P}{\text{maximize}} \sum_{i=1}^m mm(\text{map}_{\Delta x, \Delta y, \theta}(\mathbf{m}'_{P(i)}), \mathbf{m}_i) \quad (4.8)$$

where $P(i)$ is an unknown function that determines the *pairing* between **I** and **T** minutiae; in particular, each minutia has either exactly one mate in the other fingerprint or has no mate at all:

1. $P(i) = j$ indicates that the mate of the \mathbf{m}_i in **T** is the minutia \mathbf{m}'_j in **I**.
2. $P(i) = \text{null}$ indicates that minutia \mathbf{m}_i in **T** has no mate in **I**.
3. A minutia \mathbf{m}'_j in **I**, has no mate in **T** if $P(i) \neq j \forall i = 1 \dots m$
4. $\forall i = 1 \dots m, k = 1 \dots m, i \neq k \Rightarrow P(j) \neq P(k)$ or $P(i) = P(k) = \text{null}$ (this means that each minutia in **I** is associated with a maximum of one minutia in **T**, that is P is a bijective function).

Note that, in general, $P(i) = j$ does not necessarily mean that minutiae \mathbf{m}'_j and \mathbf{m}_i can be paired in the sense of Eqs. (4.6) and (4.7) but only that they are the most likely pair under the current transformation.

Expression (4.8) requires that the number of minutiae mates be maximized, independently of how strict these mates are; in other words, if two minutiae comply with Eqs. (4.6) and (4.7), then their contribution to expression (4.8) is made independently of their spatial distance and of their angular difference. Alternatives to expression (4.8) may be introduced where the residual (i.e., the spatial distance and the angular difference between minutiae) for the optimal alignment is also taken into account.

Solving the minutiae matching problem (expression 4.8) is trivial when the correct alignment $(\Delta x, \Delta y, \theta)$ is known; in fact, the pairing (i.e., the function P) can be determined by setting for each $i = 1 \dots m$:

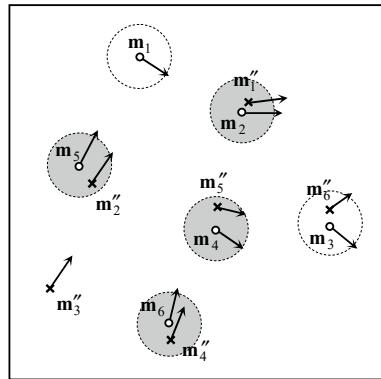


Fig. 4.6 Minutiae of **I** mapped into **T** coordinates for a given alignment. Minutiae of **T** are denoted by the “o” symbol, whereas **I** minutiae are denoted by the “x” symbol. Note that **I** minutiae are referred to as \mathbf{m}'' because what is shown in the figure is their mapping into **T** coordinates. Pairing is performed according to the minimum distance. The dashed circles indicate the maximum spatial distance. The gray circles denote successfully mated minutiae; minutia \mathbf{m}_1 of **T** and minutia \mathbf{m}''_3 of **I** have no mates, minutiae \mathbf{m}_3 and \mathbf{m}''_6 cannot be mated due to their large angular difference (Kryszczuk et al., 2004)

- $P(i) = j$ if $\mathbf{m}''_j = \text{map}_{\Delta x, \Delta y, \theta}(\mathbf{m}'_j)$ is closest to \mathbf{m}_i among the minutiae $\{\mathbf{m}''_k = \text{map}_{\Delta x, \Delta y, \theta}(\mathbf{m}'_k) \mid k = 1 \dots n, \text{mm}(\mathbf{m}''_k, \mathbf{m}_i) = 1\}$;
- $P(i) = \text{null}$ if $\forall k = 1 \dots n, \text{mm}(\text{map}_{\Delta x, \Delta y, \theta}(\mathbf{m}'_k), \mathbf{m}_i) = 0$

To comply with constraint (4) above, each minutia \mathbf{m}''_j already mated has to be marked, to avoid mating it twice or more. Figure 4.6 shows an example of minutiae pairing given a fingerprint alignment.

To achieve the optimum pairing (according to Eq. 4.8), a slightly more complicated scheme should be adopted: in fact, in the case when a minutia of **I** falls within the tolerance hyper-sphere of more than one minutia of **T**, the optimum assignment is that which maximizes the number of mates (refer to Fig. 4.7 for a simple example). Hungarian

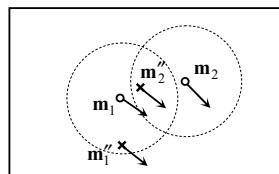


Fig. 4.7 In this example, if \mathbf{m}_1 was mated with \mathbf{m}''_2 (the closest minutia), \mathbf{m}_2 would remain unmated; however, pairing \mathbf{m}_1 with \mathbf{m}''_1 , allows \mathbf{m}_2 to be mated with \mathbf{m}''_2 , thus maximizing Eq. (4.8)

assignment algorithm (see Ahuja et al., 1991) with polynomial time complexity has been used for this purpose (see also Jea & Govindaraju, 2005; Wang et al. 2006).

The maximization in (4.8) can be easily solved if the function P (minutiae correspondence) is known; in this case, the unknown alignment $(\Delta x, \Delta y, \theta)$ can be determined in the least square sense (Chang et al., 1997; Umeyama, 1991). Unfortunately, in practice, neither the alignment parameters nor the correspondence function P is known a priori and, therefore, solving the matching problem is hard. Pasha Hosseinbor et al. (2017) derived the least square optimal alignment for two point-sets lacking one-to-one correspondence. To this purpose the assignment function P is made probabilistic: a term p_{ij} is associated to each pair of minutiae \mathbf{m}_i and \mathbf{m}'_j to denote their probability to be paired under the optimal alignment, and Eq. (4.8) is modified accordingly. An iterative optimization approach is then proposed to find the optimal alignment starting from an initial estimate of the p_{ij} 's; the initial estimate is obtained by exploiting the similarities between minutiae local structures (see Sect. 4.4).

A brute force approach, that is, evaluating all the possible solutions (correspondences and alignments) is prohibitive as the number of possible solutions is exponential in the number of minutiae (the function P is more than a permutation due to the possible null values). A few brute force approaches have also been proposed in the literature; for example, Huvanandana et al. (2000) proposed coarsely quantizing the minutiae locations and performing an exhaustive search to find the optimum alignment. He et al. (2003b) suggested a coarse-to-fine search of the discretized parameter space to determine the alignment and used Hausdorff distance to evaluate minutiae correspondences.

4.3.2 Similarity Score

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

$$\text{score} = \frac{k}{(n + m)/2} \quad (4.9)$$

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

- Minutiae quality can be used to weight differently reliable and unreliable minutiae pairs: the contribution from a pair of reliable minutiae should be higher than that from a pair where at least one the two minutiae is of low-quality (Chen et al., 2007). The quality of a minutia (and of a minutia pair) can be defined according to the fingerprint

quality in the region where the minutia lies and/or by keeping into account other local information (see Sect. 3.11.1). Cao et al. (2011) argued that minutiae belonging to minutiae clusters are less discriminative and propose to use neighbors of a minutia to measure their quality.

- The normalization in Eq. (4.9) tends to excessively penalize fingerprint pairs with partial overlap; a more effective normalization considers the number of minutiae belonging to the intersection of the two fingerprints after the optimal alignment has been determined (Jea & Govindaraju, 2005).
- Zhang et al. (2016) suggest to take unmatched minutiae (i.e., “dangling” minutiae without a mate) in due consideration when computing the matching score because they carry additional information. To this purpose, they consider the relative number of unmatched minutiae, the position similarity of unmatched minutia pairs, and the global distribution consistency of unmatched minutiae.

In general, the definition of optimal rules (heuristics) for combining various similarity contributions into a single score can be complex; some researchers (Jea & Govindaraju, 2005; Srinivasan et al., 2006; Jia et al., 2007; Feng, 2008; Lumini & Nanni, 2008) propose to apply learning-based techniques where the heuristics and its parameters are optimized to best separate genuine from impostor scores. Supervised classification is central also in the method proposed by Mansukhani et al. (2007) and by Mansukhani and Govindaraju (2008) where an SVM is trained to distinguish between genuine and false minutiae pairs. Finally, methods based on the computation of the likelihood ratio to assess the evidential value of comparisons with an arbitrary number of minutiae are quite popular in forensic identification (see Ramos et al., 2017a, b). A similarity measure derived from the likelihood ratio between the Accidental Coincident Probability and the Accidental Inconsistent Probability was introduced by Liu et al. (2010a).

In conclusion, while the definition of similarity score is very important for matching accuracy, it can be hard to evaluate its contribution independently because it is often combined with other components of matching algorithms.

4.3.3 Global Minutiae Matching Approaches

The minutiae matching problem can also be viewed as a *point pattern matching* problem. Because of its central role in many pattern recognition and computer vision tasks (e.g., object matching, remote sensing, camera calibration, motion estimation), point pattern matching has been extensively studied yielding families of approaches, hereafter introduced.

- *Hough transform*: the generalized Hough transform-based approach (Ballard, 1981; Stockman et al., 1982) converts point pattern matching to the problem of detecting peaks in the Hough space of transformation parameters. It discretizes the parameter space and accumulates evidence (or votes) in the discretized space by looking at the number of minutiae that can be paired under the different transformations. A hierarchical Hough transform-based algorithm may be used to reduce the size of the accumulator array by using a multi-resolution approach. A more in-depth analysis of this method is provided in Sect. 4.3.4.
- *Consensus-based approaches*: RANSAC (RANdom Sample Consensus) is a family of techniques largely used in computer vision and pattern recognition for model fitting in presence of outliers (Fischler & Bolles, 1981) that can be used for minutiae matching as well. Hypotheses of possible alignments are made based on the selection of a few minutiae (typically one pair from \mathbf{I} and one pair from \mathbf{T}). The remaining minutiae express their consensus for each alignment, and at the end of the process the most voted transformation is chosen. The main advantage with respect to Hough-based methods is that here no discretization of the transformation space is required, and usually a smaller number of alignments needs to be checked. *Relaxation* is another approach heavily relying on minutiae consensus. Section 4.3.5 discusses some consensus-based approaches in more details.
- *Graph matching*: point pattern matching can be turned into a graph optimization problem (Leordeanu & Hebert, 2005) by constructing a graph whose nodes are point correspondences (e.g., (i, j)) and the link connecting two nodes encode the correspondences agreement (e.g., agreement between (i, j) and (h, k)). Correct correspondences are likely to establish links among each other and thus form a strongly connected cluster. The correct assignments can be then recovered by extracting the main cluster in the graph: this can be done by using the principal eigenvectors of the graph weighted adjacency matrix (i.e., *spectral* techniques). Spectral graph matching has been successfully applied to minutiae matching by Fu et al. (2012, 2013).
- *Energy minimization with Metaheuristics*: these methods define a function that associates an *energy* or *fitness* with each solution of the problem. Optimal solutions are then derived by minimizing the energy function (or maximizing fitness) by using a stochastic algorithm such as Genetic algorithm (Le et al., 2001; Tan & Bhanu, 2006; Sheng et al., 2009), Memetic algorithm (Sheng et al., 2007), Simulated annealing (Starink & Backer, 1995), or Ant Colony optimization (Cao et al., 2012b). In general, the methods belonging to this category tend to be slow and are unsuitable for real-time minutiae matching.
- *Mapping minutiae to fixed-length representations*: mapping minutiae information to a fixed-length (ordered) vector is very appealing because after the transformation the matching becomes a simple distance computation (e.g., Euclidian, cosine, Hamming), and the design of biometric encryption technique can be greatly simplified. However, extracting invariant fixed length representations without scarifying discriminability is

challenging. It is worth noting that invariant representation here addressed are those derived by minutiae information only, in contrast with other recently introduced fixed-length deep representations extracted from the raw fingerprint pattern (Sect. 4.6.5). Minutiae need to be grouped to derive a fixed-length representation: in the early work by Willis and Myers (2001) the number of minutiae falling inside the cells of a “dart board” pattern of wedges and rings are used for the encoding; more recent approaches are based on minutiae local structures which are grouped by bag-of-words techniques (Bringer & Despiegel, 2010; Vij & Namboodiri, 2014), kernel transformations (Jin et al., 2016), and clustering (Kho et al., 2020). The *spectral minutiae representation* (in the Fourier domain) introduced by Xu et al. (2009a) is particularly intriguing; this technique, not to be confused with spectral graph matching, is briefly introduced in Sect. 4.3.6. Finally, more details on other fixed-length encoding are provided in Chap. 9 because of their relevance for fingerprint matching in the encrypted domain.

4.3.4 Hough Transform-Based Approaches

Ratha et al. (1996b) proposed a generalized Hough transform-based minutiae matching approach, whose underlying alignment transformation, besides displacement and rotation, also includes scale. The space of transformations consists of quadruples $(\Delta x, \Delta y, \theta, s)$, where each parameter is discretized (denoted by the symbol⁺) into a finite set of values:

$$\Delta x^+ \in \{ \Delta x_1^+, \Delta x_2^+, \dots \Delta x_a^+ \}, \Delta y^+ \in \{ \Delta y_1^+, \Delta y_2^+, \dots \Delta y_b^+ \}$$

$$\theta^+ \in \{ \theta_1^+, \theta_2^+, \dots \theta_c^+ \}, s^+ \in \{ s_1^+, s_2^+, \dots s_d^+ \}$$

A four-dimensional array \mathbf{A} , with one entry for each of the parameter discretization, is initially reset, and the following algorithm is used to accumulate evidence/votes:

```

for each  $\mathbf{m}_i$ ,  $i = 1 \dots m$ 
for each  $\mathbf{m}'_j$ ,  $j = 1 \dots n$ 
  for each  $\theta^+ \in \{ \theta_1^+, \theta_2^+, \dots \theta_c^+ \}$ 
    if  $ad(\theta'_j + \theta^+, \theta_i) < \theta_0$  // the minutiae angles after the rotation are sufficiently close as per Equation (4.7)
      for each  $s^+ \in \{ s_1^+, s_2^+, \dots s_d^+ \}$ 
         $\left\{ \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \end{bmatrix} - s^+ \cdot \begin{bmatrix} \cos \theta^+ & -\sin \theta^+ \\ \sin \theta^+ & \cos \theta^+ \end{bmatrix} \begin{bmatrix} x'_j \\ y'_j \end{bmatrix} \right\}$  // the map function including scale
         $\Delta x^+, \Delta y^+ = \text{quantization of } \Delta x, \Delta y \text{ to the nearest bin}$ 
         $\mathbf{A}[\Delta x^+, \Delta y^+, \theta^+, s^+] = \mathbf{A}[\Delta x^+, \Delta y^+, \theta^+, s^+] + 1$ 

```

At the end of the accumulation process, the best alignment transformation $(\Delta x^*, \Delta y^*, \theta^*, s^*)$ is obtained as

$$(\Delta x^*, \Delta y^*, \theta^*, s^*) = \arg \max_{\Delta x^+, \Delta y^+, \theta^+, s^+} A[\Delta x^+, \Delta y^+, \theta^+, s^+]$$

and the minutiae pairing is performed as previously explained (in Sect. 4.3.1). To increase robustness of the Hough transform, it is common to cast a vote not only in the discretized bin, but also in its nearest neighbors; hence, in the above pseudo-code, the accumulator update can be substituted by a simple procedure that updates all the entries in the neighborhood of the selected bin.

This algorithm, whose complexity is $O(m \times n \times c \times d)$, lends itself to parallel implementations; Ratha et al. (1995) designed a dedicated hardware consisting of a Field Programmable Gate Array (FPGA)-based point pattern matching processor.

Liu et al. (2004) proposed a hierarchical Hough transform implementation aimed at improving efficiency and accuracy of the conventional Hough approach. The discretization of the transformation parameters is done hierarchically (coarse to fine), and specific mechanisms are adopted to avoid: (i) loosing minutiae pairs due to quantization errors and (ii) matching each minutia more than once.

4.3.5 Consensus-Based Approaches

One of the first RANSAC-like approach was introduced by Udupa et al. (2001) who significantly improved an idea earlier published by Weber (1992). No scale change is allowed (rigid transformation), and the algorithm is simpler than Hough transform-based methods. The main steps are as follows:

1. The segments identified by pairs of minutiae $\overline{\mathbf{m}_{i2}\mathbf{m}_{i1}}$ in \mathbf{T} and $\overline{\mathbf{m}'_{j2}\mathbf{m}'_{j1}}$ in \mathbf{I} are considered, and from each pair of segments that have approximately the same length (remember that the transformation is rigid), the alignment parameters $(\Delta x, \Delta y, \theta)$ are derived.
2. For each alignment $(\Delta x, \Delta y, \theta)$ obtained in Step 1, \mathbf{T} and \mathbf{I} are superimposed and the pairing between the remaining minutiae is determined by using tolerance boxes, resulting in a number of mated pairs.
3. The top 10 alignments (i.e., those giving the 10 largest number of mates) are checked for consistency; in case of matching fingerprints, a majority of these alignments are mutually consistent, whereas for non-matching fingerprints they are not. A score is computed based on the fraction of mutually consistent transformations.

The final score is determined by combining the maximum number of mated pairs, the fraction of mutually consistent alignments, and the topological correspondence (i.e., minutiae angle and ridge counts) for the top 10 alignments. To reduce the computational complexity of their methods, Udupa et al. (2001) suggest to consider only segments whose lengths lie in a given range, and to filter out most of the candidate alignments in Step 1 early on according to the consistency of minutiae angles and of the ridge counts along the two segments.

Carvalho and Yehia (2004), in order to further speed-up Udupa et al. (2001) algorithm, proposed to independently pre-sort the minutiae in \mathbf{T} and \mathbf{I} , by using the probability of the lengths formed by minutiae pairs as a key to the sorting operation. This increases the chance that corresponding line segments in two fingerprints are tested in the early iterations. Bhowmick and Bhattacharya (2004) approach also relies on the above Steps 1 and 2, but the authors make use of spatial data structures such as AVL and Kd-Tree to efficiently search and match distances and points and consider minutiae quality during matching.

A more sophisticated and better performing approach was proposed by Chang et al. (1997), where both RANSAC-like model selection and Hough voting are used. Their approach consists of the following steps:

1. Detect the minutiae pair (called the *principal pair*) that receives the maximum *Matching Pair Support* (MPS) and the alignment parameters (θ , s) that can match most minutiae between \mathbf{T} and \mathbf{I} .
2. The remaining minutiae mates (i.e., the function P) are then determined once the two fingerprints have been registered to superimpose the minutiae constituting the principal pair.
3. The exact alignment is computed in the least square sense once the correspondence function is known.

To accomplish Step 1, which is at the core of this approach, the algorithm considers segments defined by pairs of minutiae $\overline{\mathbf{m}_{i2}\mathbf{m}_{i1}}$ in \mathbf{T} and $\overline{\mathbf{m}'_{j2}\mathbf{m}'_{j1}}$ in \mathbf{I} and derives, from each pair of segments, the parameters θ and s simply as

$$\theta = \text{angle}(\overrightarrow{\mathbf{m}_{i2}\mathbf{m}_{i1}}) - \text{angle}(\overrightarrow{\mathbf{m}'_{j2}\mathbf{m}'_{j1}}) \quad (4.10)$$

$$s = \frac{\text{length}(\overrightarrow{\mathbf{m}_{i2}\mathbf{m}_{i1}})}{\text{length}(\overrightarrow{\mathbf{m}'_{j2}\mathbf{m}'_{j1}})} \quad (4.11)$$

A transformation $(\Delta x, \Delta y, \theta, s)$, which aligns the two segments, must necessarily involve a scale change by an amount given by the ratio of the two segment lengths, and a rotation by an angle equal to the difference between the two segment angles (see Fig. 4.8).

The principal pair and the parameters (θ^*, s^*) are determined as

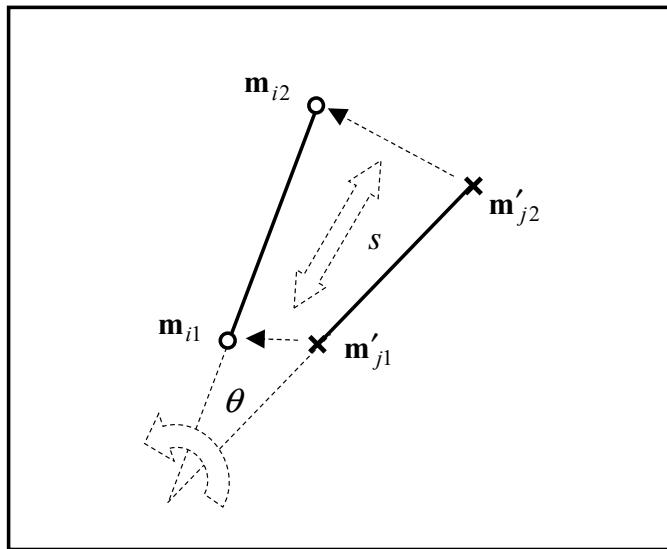


Fig. 4.8 The transformation that aligns the two segments involves a rotation and a scale change as defined by Eqs. (4.10) and (4.11)

The principal pair and the parameters (θ^*, s^*) are determined as

```

maxMPS = 0           // maximum Matching Pair Support
for each  $\mathbf{m}_{il}$ ,  $i1 = 1 \dots m$ 
for each  $\mathbf{m}'_{j1}$ ,  $j1 = 1 \dots n$  //  $\mathbf{m}_{il}, \mathbf{m}'_{j1}$  is the current pair for which MPS has to be estimated
{ Reset  $\mathbf{A}$            // the accumulator array
  for each  $\mathbf{m}_{i2}$ ,  $i2 = 1 \dots m$ ,  $i2 \neq i1$ 
  for each  $\mathbf{m}'_{j2}$ ,  $j2 = 1 \dots n$ ,  $j2 \neq j1$ 

  {  $\theta, s$  are computed from  $\overline{\mathbf{m}_{i2}\mathbf{m}_{il}}$ ,  $\overline{\mathbf{m}'_{j2}\mathbf{m}'_{j1}}$  according to Equations (4.10) and (4.11)
     $\theta^*, s^*$  = quantization of  $\theta, s$  to the nearest bins
     $\mathbf{A}[\theta^*, s^*] = \mathbf{A}[\theta^*, s^*] + 1$ 
  }

   $MPS = \max_{\theta^*, s^*} \mathbf{A}[\theta^*, s^*]$ 
  if  $MPS \geq maxMPS$ 
  {  $maxMPS = MPS$ 
     $(\theta^*, s^*) = \arg \max_{\theta^*, s^*} \mathbf{A}[\theta^*, s^*]$ 
    Principal pair = ( $\mathbf{m}_{il}, \mathbf{m}'_{j1}$ )
  }
}
}

```

Some heuristics were introduced by Chang et al. (1997) to reduce the number of segments considered and therefore, to limit the computational complexity. An example of minutiae matching by the above method is shown in Fig. 4.9.

Another interesting approach is based on relaxation (e.g., Rosenfeld & Kak, 1976; Ton & Jain 1989; Ranade & Rosenfeld, 1993): it iteratively adjusts the confidence level of each corresponding pair of points based on its consistency with other pairs until a certain criterion is satisfied. At each iteration r , the method computes $m \times n$ probabilities p_{ij} (probability that point i corresponds to point j):

$$p_{ij}^{(r+1)} = \frac{1}{m} \sum_{h=1}^m \left[\max_{k=1 \dots n} \left\{ c(i, j; h, k) \cdot p_{ij}^{(r)} \right\} \right], i = 1 \dots m, j = 1 \dots n, \quad (4.12)$$

where $c(i, j; h, k)$ is a compatibility measure between the pairing (i, j) and (h, k) , which can be defined according to the consistency of the alignments necessary to map point j into i and point k into h . Equation (4.12) increases the probability of those pairs that receive substantial support by other pairs and decreases the probability of the remaining ones. At convergence, each point i may be associated with the point j such that $p_{ij} = \max_s \{p_{is}\}$, where s is any other point in the set. The iterative refinement of p_{ij} 's is also performed by Pasha Hosseini et al. (2017) whose technique relies on a probabilistic least square optimization. The main drawback of these approaches is that a reasonable initialization of the

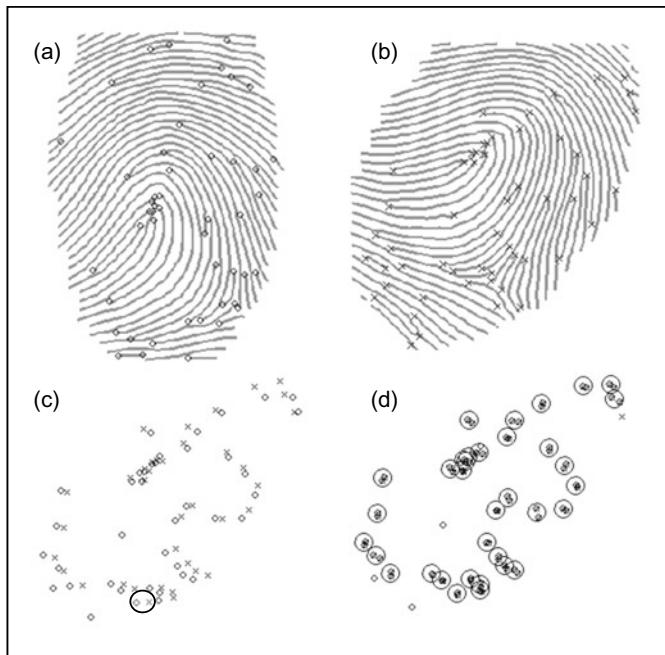


Fig. 4.9 Minutiae matching by the Chang et al. (1997) approach. Figures **a** and **b** show the minutiae extracted from the template and the input fingerprint, respectively; **c** the minutiae are coarsely superimposed and the principal pair is marked with an ellipse; **d** each circle denotes a pair of minutiae as mated by the Step 3 of the algorithm

p_{ij} 's is required for a good convergence. Therefore, such techniques find ideal application in the refinement or consolidation of an initial coarse matching (see Sect. 4.4.4).

4.3.6 Spectral Minutiae Representation

Xu et al. (2009a, b) proposed to encode fingerprint minutiae through Fourier or Fourier-Mellin transform because of their invariance properties. Unlike the correlation-based techniques in the Fourier domain previously introduced (see Sect. 4.2), here the input representation is not the raw fingerprint image but is directly obtained from the minutiae. In fact, the initial signal is defined as the sum of Dirac pulses, each located in correspondence of a minutia and optionally oriented according to the minutia angle. The transformed signal is then processed as follows:

- low-pass filtering to attenuate the higher frequencies (Dirac pulses becomes Gaussian pulses).
- computation of the magnitude of the Fourier spectrum.
- sampling on a polar-logarithmic (fixed) grid.

The obtained representation (Fig. 4.10) is invariant to translation while rotation and scaling become translations along the polar-logarithmic coordinates, so a two-dimensional correlation can be used to compute fingerprint similarity. A nice property of this technique is that it does not require to apply discrete transformations (e.g., FFT) because the peculiar signal definition allows to derive an analytical expression and to evaluate it directly on the final polar-logarithmic coordinates.

A straight matching (with no need of 2D correlation) is also possible by taking the magnitude of the Fourier spectrum of the polar-logarithmic representation thus obtaining the Fourier-Melling transform (fully invariant representation). However, the authors argued that discarding the phase in the last step usually degrades accuracy and recommend the correlation-based matching.

The main drawback of the spectral minutiae representation is the low robustness in case of small overlap between fingerprint images; in this case a large number of minutiae present in one of the images may not be present in the other image, leading to relevant variations in the global spectral representation. To overcome this weakness, “localized” implementations have been proposed by Xu and Veldhuis (2009a) and Nandakumar (2012). Further variants are also discussed in Xu and Veldhuis (2009b, 2010a, b).

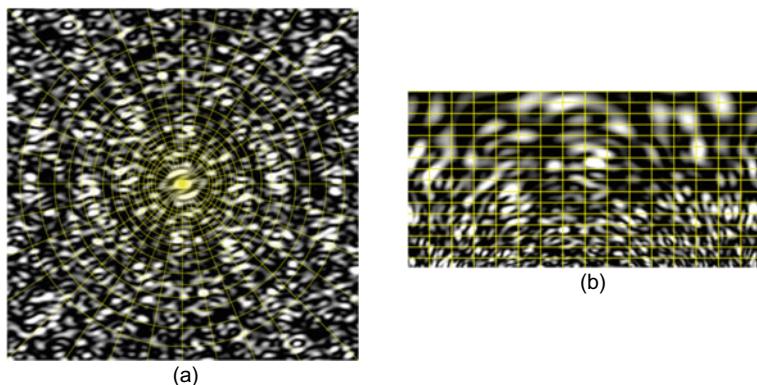


Fig. 4.10 An example of spectral minutiae representation: **a** Fourier spectrum in a Cartesian coordinate and a polar-logarithmic sampling grid; **b** Fourier spectrum sampled on a polar-logarithmic grid. © IEEE. Reprinted, with permission, from Xu et al. (2009a)

4.3.7 Minutiae Matching with Pre-Alignment

Embedding fingerprint alignment into the minutiae matching stage (as the methods presented in the previous sections do) certainly leads to the design of robust algorithms, which are often able to successfully operate with noisy and incomplete data. On the other hand, a coarse pre-alignment can help to reduce computational complexity. In theory, if a perfect pre-alignment could be achieved, the minutiae matching could be reduced to a trivial pairing. Pre-alignment can be absolute or relative.

In *absolute* pre-alignment (a.k.a. pose estimation) each fingerprint template is pre-aligned, independently of the others, before storing it in the database. Matching an input fingerprint \mathbf{I} with a set of templates requires \mathbf{I} to be independently registered just once, and the resulting aligned representation to be matched with all the templates. Pose estimation was discussed in detail in Sect. 3.5.5: unfortunately, reliable pose estimation is challenging especially for partial and low-quality fingerprints.

In *relative* pre-alignment (a.k.a. *registration*) the input fingerprint \mathbf{I} has to be pre-aligned with respect to each template \mathbf{T} in the database; $1:N$ identification requires N independent pre-alignments. Relative pre-alignment may determine a significant speed up with respect to the algorithms that do not perform any pre-alignment but cannot compete in terms of efficiency with absolute pre-alignment. However, relative pre-alignment is in general more effective (in terms of accuracy) than absolute pre-alignment because the features of the template \mathbf{T} may be used to drive the registration process. The rest of this Section focuses on relative pre-alignment, which may be carried out in the following ways:

- By superimposing the singularities (e.g., Chikkerur & Ratha, 2005).
- By correlating the orientation images.
- By comparing ridge features (e.g., length and orientation of the ridges).
- At pixel level (i.e., *dense registration*). These techniques are discussed in Sect. 4.5.4 because of their central role in distortion correction.

Yager and Amin (2006a) showed that relative pre-alignment of fingerprints based on a dynamic selection among different registration techniques lead to significant improvement over single alignment approaches.

Orientation image-based relative pre-alignment

A basic implementation requires computing the similarity between the orientation image of \mathbf{T} (here denoted by \mathbf{D}_T) with every possible transformation $\mathbf{D}_I^{(\Delta x, \Delta y, \theta)}$ of the orientation image of \mathbf{I} , where Δx , Δy denote the translation and θ the rotation, respectively. This is similar to correlating images at intensity level (refer to Sect. 4.2) even if the block-wise definition of the orientation image and the nature of the orientations require some adjustments; in particular, the similarity among angles must be defined by keeping into account

angle circularity and rotating/shifting \mathbf{D}_I may require the use of resampling techniques to avoid degrading accuracy. Unlike classical definition based on angular differences, Liu et al. (2006) define the similarity between orientation images by means of the Mutual Information (Guías, 1977). Yager and Amin (2006b), besides the orientation image, also use ridge frequency and ridge curvature images.

To determine the optimal alignment, the space of possible transformations is usually discretized and the single transformation leading to the highest similarity is selected; see for example Lindoso et al. (2007). To improve efficiency, instead of performing an exhaustive evaluation of all the transformations:

- Yager and Amin (2004, 2006b) suggest using local optimization techniques such as the steepest descent.
- Nilsson and Bigun (2005) focus on 1D projections of orientation images.
- Liu et al. (2006) implement a two-stage coarse-to-fine strategy.

Finally, Lan et al. (2019, 2020) showed that the orientation image can be useful also for non-rigid fingerprint registration; their iterative optimization approach provides, as a byproduct, an estimation of the distortion between the two fingerprints.

Ridge-based relative pre-alignment

An interesting minutiae matching approach that exploits ridge features for relative pre-alignment was proposed by Jain et al. (1997). The relative pre-alignment is based on the observation that minutiae registration can be performed by registering the corresponding ridges. In fact, each minutia in a fingerprint is associated with a ridge; during the minutiae extraction stage, when a minutia is detected and recorded, the ridge on which it resides is also recorded. The ridge is represented as a planar curve, with its origin coincident with the minutia and its x -coordinate being in the same direction as the minutia angle. The ridge matching task proceeds by iteratively matching pairs of ridges until a pair is found whose matching degree exceeds a certain threshold. The pair found is then used for relative pre-alignment.

Other implementations of ridge-based alignment can be found in Luo et al. (2000), Hao et al. (2002), He et al. (2003a), Cheng et al. (2004), and Feng and Cai (2006).

4.4 Global Versus Local Minutiae Matching

Local minutiae matching consists of comparing two fingerprints according to local minutiae structures (or *descriptors*); local structures are characterized by attributes that are invariant with respect to global transformation (e.g., translation, rotation, etc.) and therefore are suitable for matching without any a priori global alignment. Matching fingerprints based only on local minutiae arrangements relaxes global spatial relationships which are

highly distinctive and therefore reduce the amount of information available for discriminating fingerprints. Global versus local matching is a tradeoff among simplicity, low computational complexity, and high distortion-tolerance (local matching), and high distinctiveness on the other hand (global matching). Actually, the benefits of both local and global matching can be obtained by implementing hybrid strategies that perform a *local structure matching* followed by a *consolidation* stage. The local structure matching allows to quickly and robustly determine pairs of minutiae that match locally (i.e., whose neighboring features are compatible) and derive from them one or more candidate alignments for \mathbf{T} and \mathbf{I} . The consolidation is aimed at verifying if and to what extent local matches hold at global level. It is worth noting that the consolidation step is not mandatory and a score can be derived directly from the local structure matching. The local matching itself can also lead to an early rejection in case of very different fingerprints. According to the public information released by the designer of algorithms submitted to FVC-onGoing, most state-of-the-art fingerprint recognition algorithms are based on such a two-stage approach: local minutiae matching, followed by global consolidation.

Some early methods were proposed by Hrechak and McHugh (1990), Chen and Kuo (1991) and Wahab et al. (1998): local minutiae matching algorithms evolved through three generations of methods: (i) the archetype approaches by Jiang and Yau (2000) and Ratha et al. (2000) who first encoded the relationship between a minutia and its neighboring minutiae in term of invariant distances and angles and proposed global consolidation; (ii) the plethora of variants and evolutions of Jiang and Yau (2000) and Ratha et al. (2000) methods including minutiae triangles and texture-enriched local structures; (iii) Minutiae Cylinder Code (MCC) and its variants. Finally, some learning-based minutiae descriptors have been introduced for latent fingerprints, for details refer to Chap. 6.

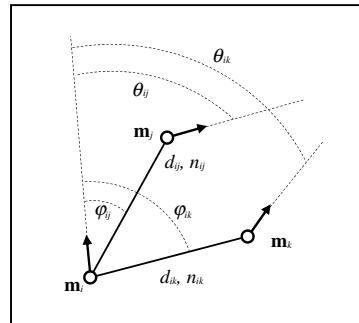
4.4.1 Archetype Methods for Nearest Neighbor-Based and Fixed Radius-Based Local Minutiae Structures

In Jiang and Yau (2000) local structures are formed by a central minutia and its two nearest-neighbor minutiae; the feature vector \mathbf{v}_i associated with the minutia \mathbf{m}_i , whose nearest neighbors are minutiae \mathbf{m}_j (the closest to \mathbf{m}_i) and \mathbf{m}_k (the second closest), has a fixed-length:

$$\mathbf{v}_i = [d_{ij}, d_{ik}, \theta_{ij}, \theta_{ik}, \varphi_{ij}, \varphi_{ik}, n_{ij}, n_{ik}, t_i, t_j, t_k]$$

where d_{ab} is the distance between minutiae \mathbf{m}_a and \mathbf{m}_b , θ_{ab} is the angular difference between the angles θ_a and θ_b of \mathbf{m}_a and \mathbf{m}_b , φ_{ab} is the angular difference between the angle θ_a of \mathbf{m}_a , and the direction of the edge connecting \mathbf{m}_a to \mathbf{m}_b , n_{ab} is the ridge count between \mathbf{m}_a and \mathbf{m}_b , and t_a is the minutia type of \mathbf{m}_a (Fig. 4.11).

Fig. 4.11 Features of the local structures used by Jiang and Yau (2000)



Local minutiae matching is performed by computing, for each pair of minutiae \mathbf{m}_i and \mathbf{m}'_j , $i = 1 \dots m$, $j = 1 \dots n$, a weighted Euclidean distance between their vectors \mathbf{v}_i and \mathbf{v}'_j . The computation of the $m \times n$ distances is very fast and an overall similarity score between \mathbf{T} and \mathbf{I} can be directly derived by the top matching (i.e., less distant) structure pairs. However, to improve accuracy, Jiang and Yau (2000) suggested implementing a consolidation step as described in Sect. 4.4.4.

The local structures introduced by Ratha et al. (2000) are characterized by a fixed radius (and a variable length). The *star* associated with the minutia \mathbf{m}_i for a given distance d_{\max} is the graph $S_i = (V_i, E_i)$ consisting of

- The set of vertices V_i containing all the minutiae \mathbf{m}_j whose spatial distance $sd()$ from \mathbf{m}_i is less than or equal to d_{\max} : $V_i = \{ \mathbf{m}_j \mid sd(\mathbf{m}_i, \mathbf{m}_j) \leq d_{\max} \}$.
- The set of edges $E_i = \{ \mathbf{e}_{ij} \}$, where \mathbf{e}_{ij} is the edge connecting minutia \mathbf{m}_i with minutia \mathbf{m}_j in V_i ; \mathbf{e}_{ij} is labeled with a 5-tuple $(i, j, sd(\mathbf{m}_i, \mathbf{m}_j), rc(\mathbf{m}_i, \mathbf{m}_j), \varphi_{ij})$, where $rc(\mathbf{m}_i, \mathbf{m}_j)$ is the ridge count between \mathbf{m}_i and \mathbf{m}_j , and φ_{ij} is the angle subtended by the edge with the x -axis.

Figure 4.12 shows the star of a given minutia for two different values of d_{\max} .

During local minutiae matching, each star from \mathbf{I} is matched against each star from \mathbf{T} . The matching between two stars $S_a = (V_a, E_a)$ from \mathbf{T} and $S_b = (V_b, E_b)$ from \mathbf{I} is performed as follows: given a starting pair of edges $\mathbf{e}_{aj} \in E_a$ and $\mathbf{e}_{bk} \in E_b$, a clockwise traversing of the two set of edges E_a and E_b is executed in increasing order of radial angles φ and a score is obtained by accumulating the similarities between pairs of corresponding edges. The traversing is repeated by using every pair of edges as starting pair, and the maximum score is returned. It is worth noting that Ratha et al. (2000) did not encode the angles φ as relative angles with respect to the angle of central minutia; for this reason, the stars are not rotation invariant and the matching between two stars is not straightforward. Ratha et al. (2000) also recommended a consolidation step (see Sect. 4.4.4).

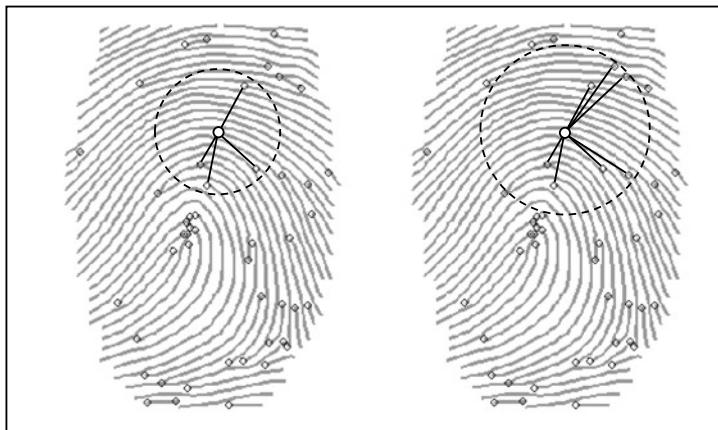


Fig. 4.12 The stars of a given minutia for $d_{\max} = 70$ (left) and $d_{\max} = 100$ (right) (Ratha et al., 2000)

4.4.2 Evolution of Local Structure Matching

The matching of local structures invariant with respect to translation and rotation forms the basis of most of the minutiae matching algorithms proposed after the year 2000. Instead of systematically describing individual approaches we prefer to focus attention on the main ideas that lead to evolutions with respect to the archetype methods described in Sect. 4.4.1.

Nearest neighbour-based structures

Jea and Govindaraju (2005) argue that a limitation of Jiang and Yau (2000) descriptors is the possibility of exchanging the nearest neighbour minutia with the second nearest neighbour in case the two distances are similar. Such a minutiae flip would lead to a failure of local matching; hence, Jea and Govindaraju (2005) propose choosing the minutiae order based on the angular relationships, which are usually more stable.

In Chikkerur et al. (2006) local structures (called K -plet) are formed by the K nearest neighbor minutiae. To avoid reducing the neighborhood size too much (this can happen for high minutiae density), a constraint can be added to ensure that the K nearest neighbors are equally distributed in the four quadrants around the minutia. Similarly, Kwon et al. (2006) subdivide the plane around a minutia into K angular sectors and take the nearest neighbour minutia for each sector. The comparison of two K -plets is performed by Chikkerur et al. (2006) by computing the edit distance between the two strings obtained by concatenating the K neighboring minutiae sorted by their radial distance by the central minutia; in spite of a higher matching complexity, this approach can tolerate, to some extent, missing and spurious minutiae.

In the method proposed by Deng and Huo (2005) the nearest neighbour minutiae are not directly selected based on Euclidean distance, but from the Delaunay triangulation of the entire minutiae set; in particular a local structure is constructed around each minutia by labelling with invariant features (the same used by Jiang and Yau (2000)) the edges that connect it with the neighbouring minutiae in the Delaunay triangulation. This results in a variable length descriptors. A similar neighbouring relationship was used in the method by Yu et al. (2005).

Fixed radius-based structures

One drawback of Ratha et al. (2000) approach is the absolute encoding of angles φ_{ij} . Most of the variants of this method encode this angle as relative to the central minutia angle; this makes the resulting feature invariant to rotation and the local structure comparison simpler. Another useful feature that was not considered for local structure matching by Ratha et al. (2000) is the angular difference between the central minutia and the neighbouring ones (these are denoted by the symbol θ in Fig. 4.11).

Matching fixed radius-based structures can lead to border (boundary) errors: in particular, minutiae which are close to the local region border in one of the two fingerprints can be mismatched because of different local distortion or location inaccuracy that cause the same minutiae to move out of the local region in the second fingerprint. To overcome this problem Chen et al. (2006b) proposed matching local structures in two steps: in the first step local structures of **T** with radius R are matched with local structures of **I** with radius $R+r$, where r is a small offset; in the second step, **T** local structures with radius $R+r$ are matched with **I** local structures of radius R . Feng (2008) suggests labelling a subset of minutiae which fall within the fixed-radius local structures as *should-be matching*; in particular, the above constraint is not enforced for: unreliable (i.e., low-quality), occluded (i.e., out of the foreground area), and close-to-the-border minutiae.

Minutiae triangles

Minutiae triangles (or triplets) were first introduced in the context of fingerprint indexing by Germain et al. (1997), as explained in Chap. 5. Although the notion of minutiae triangles was specifically introduced for fingerprint indexing (see Sect. 5.4.3), some researchers have demonstrated their potential for local structure matching for fingerprint verification; see for example, Tan and Bhanu (2003, 2006), Parziale and Niel (2004), Yin et al. (2005), Chen et al. (2006b), Zhao et al. (2006), and Xu et al. (2007). Several features invariant with respect to translation, rotation (and sometimes scale) can be extracted from triangles whose vertices coincide with minutiae:

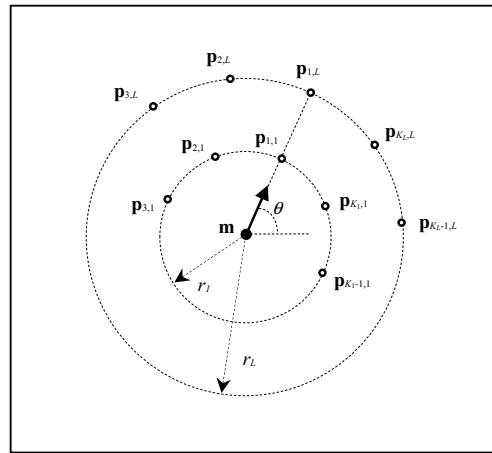
- Germain et al. (1997) used the length of each side, the angles that the ridges make with respect to the x -axis of the reference frame, and the ridge count between each pair of vertices.
- Tan and Bhanu (2003) selected the triangle features according to some stability analysis. They finally choose: triangle handedness, triangle type, triangle direction, maximum side, minimum and median angle, minutiae density in a local area and ridge counts along the three sides.
- Chen et al. (2006b) also record, for each triangle vertex, the average deviation of the local orientations in a neighbourhood.

The similarity between any two triangles can be defined according to a weighted distance between the triangles features or, as proposed by Tan and Bhanu (2003), by counting the number of minutiae that can be matched once the two fingerprints are aligned with the transformation aligning the two triangles. For local structures centred into a single minutiae, as that proposed by Jiang and Yau (2000), the number of local structures is the same as the number of minutiae; on the other hand, the number of possible minutiae triangles equals the combination of three minutiae from a set of n minutiae that is potentially very large (e.g., 19,600 triangles for $n = 50$). Some rules and constraints are then utilized to limit the number of triplets and at the same time to select the most discriminating triangles (e.g., collinear or near collinear triplets are excluded and the triangle sides must have a minimum length). Parziale and Niel (2004), Yin et al. (2005), and Xu et al. (2007) suggested using only the triangles resulting from the Delaunay triangulation of the minutiae set; in fact, the Delaunay triangulation was found to have the best structural stability under random positional perturbations (Bebis et al., 1999). To further improve tolerance to perturbation, Liang et al. (2007) suggest using both 0-order and 1-order Delaunay triangles.

Texture-based local structures

Tico and Kuosmanen (2003) proposed to create minutiae-centered local structures by using invariant information not directly related to the neighbouring minutiae. Their local descriptors include orientation information taken at some sampling points around a minutiae \mathbf{m} (see Fig. 4.13). Each orientation is recorded as relative angle with respect to the minutiae angle θ and therefore it is invariant with respect to the minutiae location and angle. The sampling points \mathbf{p} are located along L concentric circles of radii r_l , ($1 \leq l \leq L$); the first sampling point along each circle has a zero angular displacement with respect to the minutiae angle; the rest of the K_l sampling points are uniformly distributed; note that the number of sampling points varies for the different circles. Tico and Kuosmanen (2003) proposed the relation $K_l = \lceil 172 \cdot r_l / R \rceil$, where R is the fingerprint resolution in dpi. The similarity between two descriptors can be simply computed as the average angular difference between corresponding orientation information. It is worth noting that using local structures not encoding the inter-minutiae relationships can better tolerate missing

Fig. 4.13 The sampling point $\mathbf{p}_{i,j}, j = 1 \dots L, i = 1 \dots K_j$ around a minutia \mathbf{m} with angle θ as proposed by Tico and Kuosmanen (2003)



and false minutiae; however, a subsequent consolidation step is, in this case, mandatory to exploit the discriminability given by the minutiae location and angles.

Several variants of Tico and Kuosmanen descriptors have been proposed; the main differences can be summarized as follows:

- Diverse definitions of the sampling pattern; see for example, Qi and Wang (2005), Zhu et al. (2005) and usage of average local orientation over blocks (Wang et al., 2007) instead of point-wise orientations.
- Sampling (at each point \mathbf{p}) of further features, such as the local ridge frequency (Feng, 2008), the Linear Symmetry (Nilsson & Bigun, 2001), the gray-scale variance (He et al. 2006, 2007), the responses of Gabor filters (Chikkerur et al., 2006; Benhammadi et al. 2007a; Hu et al., 2017; Alshehri et al., 2018), and binary gradient patterns (Alshehri et al., 2018).
- Combination with classical structures based on minutiae inter-relationship such as Jiang and Yau (2000); see for example, Wei et al. (2006) and Feng (2008).
- Combination with local ridge information such as distances, relative angles and curvatures sampled at different points along the ridge originating at each minutia, different kinds of ridge-counts: Ng et al. (2004), Sha and Tang (2004), Tong et al. (2005), Sha et al. (2006), Feng et al. (2006), He et al. (2006, 2007), Wang et al. (2007) and Zhang et al. (2007), Cao et al. (2012a), Hu et al. (2017).

4.4.3 Minutiae Cylinder Code

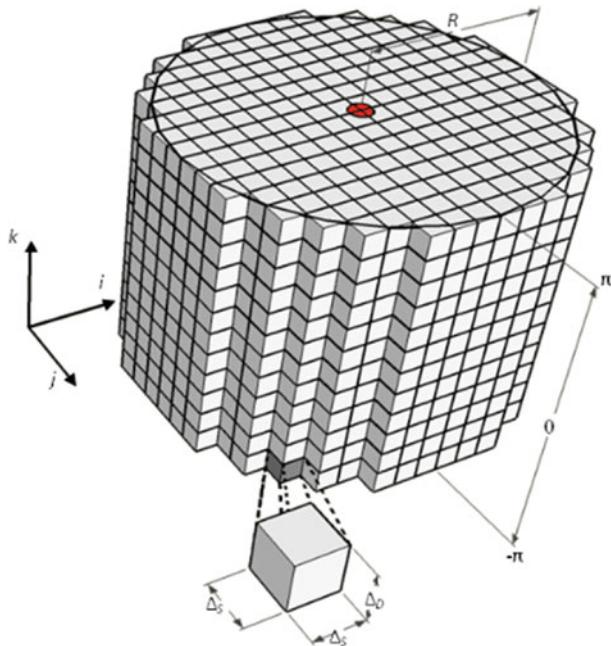
The Minutiae Cylinder-Code representation (MCC) was introduced by Cappelli et al. (2010a) to capture the advantages of both nearest-neighbor and fixed-radius approaches and avoid their weaknesses. As nearest-neighbor local structures MCC is a fixed-length (order invariant) representation and therefore can be matched very efficiently; as fixed-radius local structures MCC tolerates missing and spurious minutiae; furthermore, it deals, by design, with border minutiae and trespassing in the background region.

The local descriptor associated to each minutia \mathbf{m} encodes spatial and directional relationships between \mathbf{m} and its neighborhood minutiae and can be conveniently represented as a cylinder, whose base and height are related to the spatial and directional information, respectively (Fig. 4.14). The cylinder is divided into sections: each section corresponds to an angular difference in the range $[-180^\circ, +180^\circ]$; sections are discretized into cells.

During the creation of a cylinder, a numerical value is calculated for each cell, by accumulating contributions from minutiae in a neighborhood. The contribution of each minutia \mathbf{m}_i to a cell (of the cylinder corresponding to a given minutia \mathbf{m}) depends both on:

- spatial information (how much \mathbf{m}_i is close to the center of the cell), and
- directional information (how much the angular difference between \mathbf{m}_i and \mathbf{m} is similar to the angular difference associated to the section where the cell lies).

Fig. 4.14 A graphical representation of the discretized cylinder associated to a given minutia



In other words, the value of a cell represents the likelihood of finding minutiae that are spatially close to the cell center and whose directional difference with respect to \mathbf{m} is similar to a given value; Figure 4.15 shows the cylinder associated to a minutia with five minutiae in its neighborhood. It is worth noting that a cylinder is invariant for translation and rotation, since: (i) it only encodes distances and angular differences between minutiae, and (ii) its base is rotated according to the corresponding minutia direction.

Matching two fingerprints with MCC requires to match all-against-all the cylinders associated to the minutiae of both fingerprints to find out a set of correspondences that can be further consolidated at global level. However, due to the fixed-length nature of the representation and the possibility of quantizing cell values with single bits (with just a minor accuracy drop), bitwise operators (XOR) can be used for distance computation thus achieving very high throughput. In 2015, Cappelli et al. (2015) showed that a GPU-based MCC optimization can match more than 40 million fingerprints per second; five years

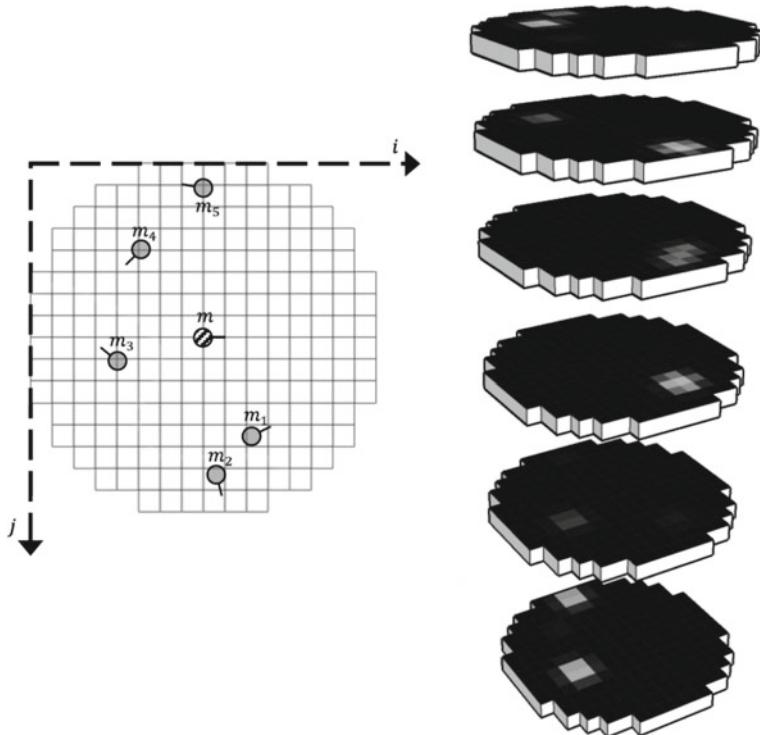


Fig. 4.15 In this example the cells of a minutia cylinder (with 6 sections) are lightened according to the position and angles of the 5 minutiae falling in the neighborhood of \mathbf{m} . Note that each neighboring minutia contribution is not limited to a single cell but is maximum in a given cell and smoothly fades along both the spatial and angular dimension: this increases robustness against small inaccuracies in the minutiae locations and angles

later (2020) such throughput increased to about 115 million on a single Titan RTX GPU. Further details on MCC implementation, parameter tuning and optimization can be found in Cappelli et al. (2010b, 2018), and Sutarno and Kistijantoro (2017).

MCC was demonstrated to be more accurate than previous local descriptors in the study by Feng and Zhou (2011) whose comparative evaluation focuses on four categories of fingerprints: good quality, poor quality, small common region, and large plastic distortion; MCC achieved the best performance in three of the four categories, and texture-based descriptor performed better for the small common region category.

Some MCC variants have been proposed which: (i) exploit local quality (Izadi et al., 2012); (ii) take scale into consideration (Zang et al., 2013); (iii) anchor the cylinder definition to minutiae pairs (Abe & Shinzaki, 2015). Furthermore, a global MCC implementation was introduced by Luo et al. (2014) and MCC descriptors are used by Paulino et al. (2014) for global latent to rolled fingerprint alignment.

4.4.4 Consolidation

Although the scores obtained from the comparison of local structures can directly lead to a final matching decision, usually a further consolidation stage is implemented to check whether the local similarity holds at the global level. In fact, some minutiae-based local configurations could coincide in fingerprints from different fingers, but the chance that their spatial relationships are coherent is markedly lower. Even though different consolidation techniques have been proposed in conjunction with a specific local structure matching, we believe that most of the underlying ideas can be cross-implemented, and for this reason in this section we concentrate only on the consolidation step.

A conceptually simple consolidation approach is to use a global minutiae matching technique (e.g., one of those introduced in Sect. 4.3) where the minutiae pairs ($\mathbf{m}_i, \mathbf{m}'_j$) are restricted to those obtaining a sufficient score at local level. This can significantly reduce the search space and make the entire matching very efficient. However, if the global matching algorithm is not distortion tolerant, such a consolidation could reduce the overall robustness of the approach.

Single versus Multiple transformations

The simplest consolidation approach relies on the alignment of **T** and **I** minutiae based on the *best transformation* resulting from the local structure matching:

- In Jiang and Yau (2000) the best transformation is that obtained by aligning the central minutiae of the two local structures receiving the highest matching score: the translation vector is the displacement between the minutiae origin and the rotation is the angular difference between the minutiae angles.

- In Tong et al. (2005) the best transformation is determined by the simultaneous local matching of two minutiae in **T** with two minutiae in **I**. This is computationally more demanding but can lead to a more robust alignment.

Once **T** and **I** minutiae have been aligned, the pairing can be computed by using tolerance boxes.

Using a single (the best) transformation for the alignment can lead to unsatisfactory results on low-quality and distorted fingerprints. In fact, even for genuine pair of fingerprints, the best transformation originating by local structures matching is not necessarily the best transformation that maximizes pairing at the global level. Therefore, several authors have proposed to adopt multiple candidate transformations for the alignments and:

- Select the final transformation according to the highest score achieved (i.e., max. rule) in the final pairing stage; see for example, Parziale and Niel (2004), Deng and Huo (2005), Wei et al. (2006), Zhang et al. (2007), He et al. (2007), Wang et al. (2007), and Feng (2008).
- Restrict the global matching to regions close to each reference pair (Lee et al., 2002).
- Fuse the results of multiple registrations. Sha et al. (2006) suggest a multiple transformation-based consolidation where the subsets of minutiae corresponding to the top alignments, i.e., those individually leading to the maximum number of matching minutiae, are fused. In the approach by Zhao et al. (2006), minutiae are a-priori clustered and global matching is performed among clusters starting from the set of plausible alignments resulting from local matching. Finally, Chen et al. (2003) perform a tessellation of the overlapping area in hexagonal regions and combine the optimal alignments of these regions.

Consensus of transformations

The aim is to compute to what extent the individual transformations resulting from the local structure matching are consistent and/or to extract a maximal subset of cross-consistent transformations. This can be obtained by binning (or clustering) the transformation space. Ratha et al.'s (2000) consolidation checks whether the TOP matching local structures are consistent: a pair of local structures is consistent if their spatial relationships (distance and ridge count) with a minimum fraction of the remaining structures in TOP are consistent. Other examples can be found in Chen et al. (2003), Jea and Govindaraju (2005), and Kwon et al. (2006).

Zhang et al. (2003) and He et al. (2006) create a histogram starting from the scores of the local structures and the corresponding transformations and selected the optimum alignment in correspondence with the histogram peak. The histogram technique is conceptually not much different from the voting process underlying a Hough transform-based

approach. A similar but more elegant approach, based on Parzen-window estimation, was proposed by He et al. (2007).

Feng et al. (2006) and Cappelli et al. (2010a) consolidate local structure compatibility based on a relaxation process (see Eq. 4.12). The matching scores are iteratively updated based on the scores of neighbouring local structures. If A and B are two neighbouring local structures in \mathbf{T} and C and D are two local structures in \mathbf{I} , then the compatibility coefficient $c(A, C, B, D)$, that is the support given by B, D to the pair A, C is determined according to the compatibility of the two transformations aligning B, D and A, C . After a number of iterations, the scores of the genuine local structure pairs, which are consistent with their neighbours, are substantially incremented and can be more easily discriminated from the impostor pairs that obtained an initial high score by chance.

Incremental consolidation

Chikkerur et al. (2006) arrange their local structures (see K -plet definition in the Sect. 4.4.2) into a directed graph, whose nodes are the K -plets and whose connectivity (i.e., edges) encode the K nearest neighbour relationships between the nodes (see Fig. 4.16). The matching between two graphs is performed through a dual graph traversal algorithm, that, starting from a given pair of nodes $\langle \mathbf{u}, \mathbf{v} \rangle$ (\mathbf{u} from \mathbf{T} and \mathbf{v} from \mathbf{I}),

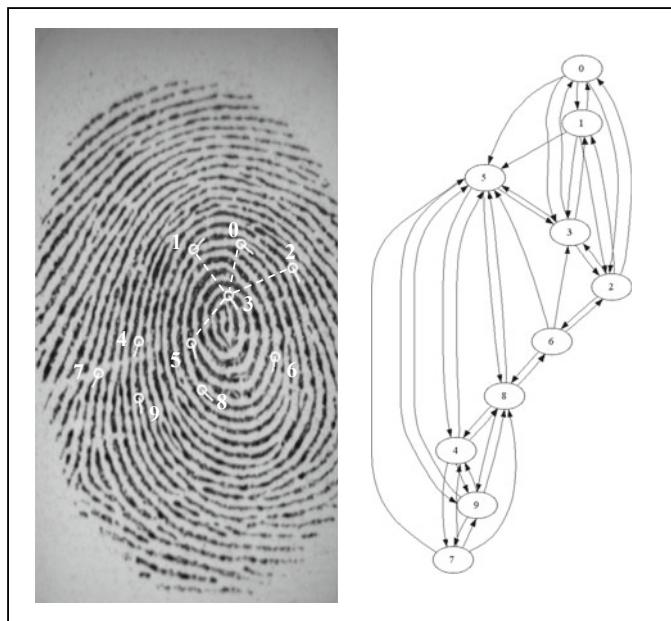


Fig. 4.16 Chikkerur et al. (2006): the graphs built from the K -plets of a fingerprint image; the K -plet of the node 3 is highlighted over the fingerprint image. © Springer Nature. Reprinted, with permission, from Chikkerur et al. (2006).

propagates the visit to neighbouring nodes (in the K -plet) in a breadth-first fashion; if two nodes cannot be locally matched, the visit along the corresponding branch is terminated. At the end of the visit, the algorithm returns the number of matched nodes. Since no point correspondence is known a-priori, the whole dual graph traversal is repeated for every pair $\langle u, v \rangle$ of nodes and the best solution (i.e., the maximum number of matched nodes) is finally chosen. A similar approach is described in Jain et al. (2006).

Xu et al. (2007) method uses triangles as local structures, and its consolidation is then based on the following steps: (i) a *growing region* is built around each triangle by extending it with neighbouring triangles; (ii) for each pair of triangles (one from T and the other from I) the matching score (called *credibility*) is consolidated according to the number of matching minutiae in the two corresponding growing regions; (iii) all growing regions are fused into a *fusion region*: the fusion process is driven by a majority voting with competition strategy, in which every pair of minutiae structures votes for the other and only the minutiae structure pairs that accumulated enough votes survive and are finally fused. The strength of the votes depends on the compatibility between the structures and the credibility of the voter. Other triangle based consolidation techniques were introduced by Kovacs-Vajna (2000), Qi and Wang (2005), and Feng et al. (2006).

4.5 Dealing with Distortion

Non-linear distortion introduced during fingerprint sensing is certainly one of the most critical intra-class variability. NIST Special Database 24 (Watson, 1998) contains videos of live-scan fingerprint data that clearly show the effect of distortion produced by users deliberately moving their fingers on the scanner surface once they are in contact. Figure 4.17 shows some examples of distorted fingerprints from FVC2004 DB1.

Ratha and Bolle (1998) demonstrated that equipping a fingerprint reader with a mechanical force sensor may help in controlling fingerprint acquisition and guiding the users toward a correct interaction with the reader. Dorai et al. (2000, 2004) proposed an automated method for detecting the presence of distortion from compressed fingerprint videos and rejecting the distorted frames. Unfortunately, most of the commercial acquisition devices do not mount force sensors and are not able to deliver images at a high frame rate and, therefore, the need for distortion-tolerant matchers is apparent.

Some of the matching algorithms discussed earlier incorporate ad hoc countermeasures to deal with distortion; in particular, local structure matching (Sect. 4.4) is itself a valid strategy to control the effects of distortion, while registration (Sect. 4.3.7) and consolidation (Sect. 4.4.4) techniques are often designed to explicitly cope with distortion.

Throughout the rest of this section we first review the proposed distortion models and then discuss the techniques that were explicitly introduced to address the problem of fingerprint matching under non-linear distortion. Once again, we attempt to consistently group them according to the underlying ideas.

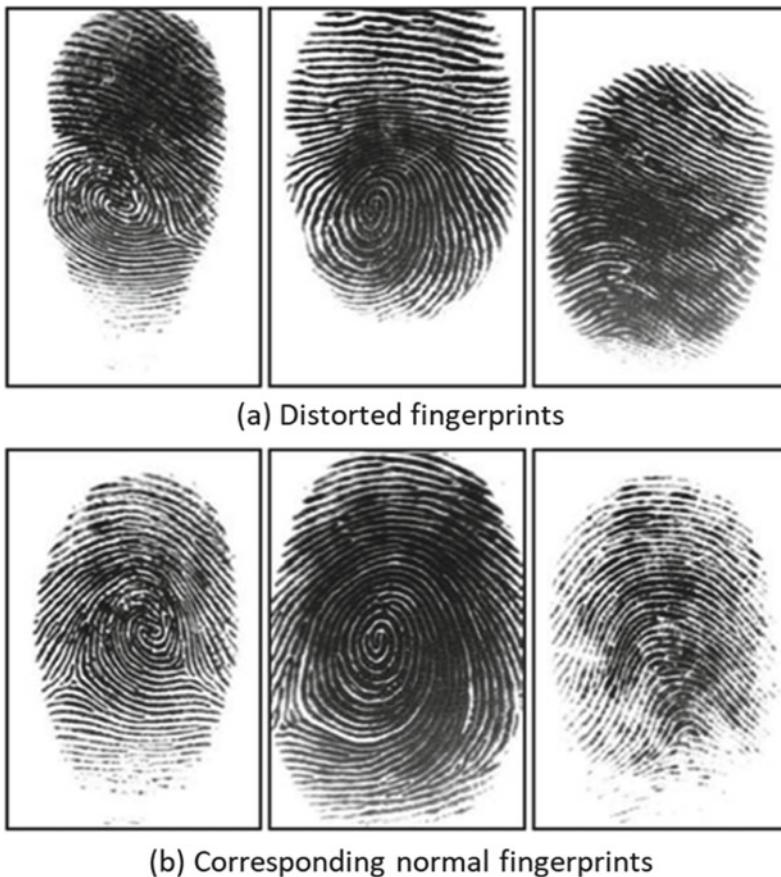


Fig. 4.17 Examples of distorted fingerprints (a) and their corresponding undistorted versions (b) from FVC2004 DB1

4.5.1 Fingerprint Distortion Models

Distortion models have been proposed to describe how a fingerprint pattern can be modified when a finger touches a rigid flat surface. Understanding such mechanisms not only can lead to the design of more robust matchers but also allows simulating distortion for fingerprint synthesis (see Chap. 7).

Cappelli et al. (2001) explicitly modelled skin distortion caused by non-orthogonal pressure of the finger against the sensor surface. By noting that the finger pressure against the sensor is not uniform but decreases from the center towards the borders, their distortion model defined three distinct regions (see Fig. 4.18):

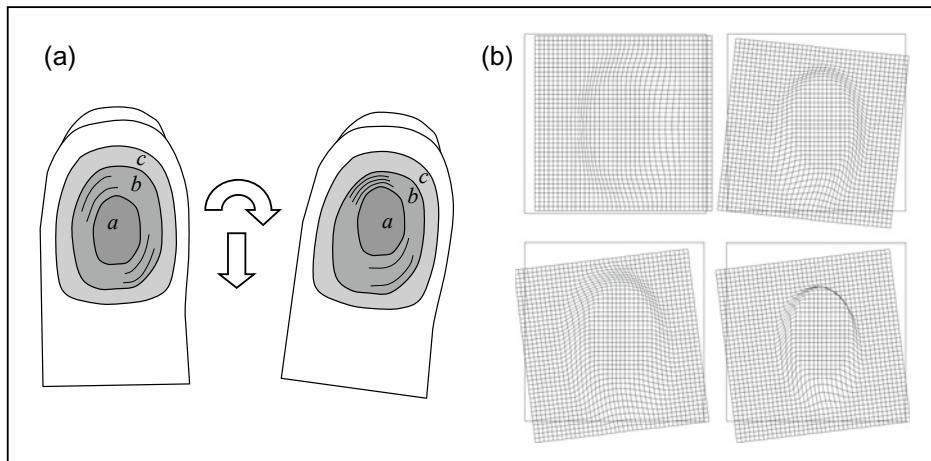


Fig. 4.18 **a** Bottom view of a finger before and after the application of traction and torsion forces. In both cases the fingerprint area detected by the sensor (i.e., the finger touching area) is delimited by the external boundary of region *c*. **b** Distortions of a square mesh obtained by applying the above model with different parameter settings. The black square denotes the initial mesh position and its movement with respect to the mesh boundary indicating the amount of displacement and rotation that occurred. © Springer Nature. Reprinted, with permission, from Cappelli et al. (2001).

1. A close-contact region (*a*), where the high pressure and the surface friction do not allow any skin slippage.
2. An external region (*c*), whose boundary delimits the fingerprint visible area, where the small pressure allows the finger skin to be dragged by the finger movement.
3. A transitional region (*b*) where an elastic distortion is produced to smoothly combine regions *a* and *c*. The skin compression and stretching is restricted to region *b*, as points in *a* remain almost fixed and points in *c* rigidly move together with the rest of the finger.

The distortion model is defined by a mapping $\mathbb{R}^2 \rightarrow \mathbb{R}^2$ that can be viewed as an affine transformation (with no scale change) which is progressively “braked” as it moves from *c* towards *a*. Each point \mathbf{v} is mapped into $distortion(\mathbf{v})$ such that

$$distortion(\mathbf{v}) = \mathbf{v} + \Delta(\mathbf{v}) \cdot brake(shapedist_a(\mathbf{v}), k) \quad (4.13)$$

where $\Delta()$ specifies the affine transformation of a point in the external region *c*; $shapedist_a()$ is a shape function describing the boundary of region *a*; $brake$ is a monotonically increasing function that controls the gradual transition from region *a* towards region *c*; the

input parameter k regulates the skin plasticity. Figure 4.18b shows some examples of distortion by varying the parameters. This model was used to simulate fingerprint distortion in the SFinGe generation method described in Chap. 7.

A distortion model, similar to that proposed by Cappelli et al. (2001), was introduced by Novikov and Ushmaev (2004). This model is based on the solution of a set of Navier linear partial differential equations (see Sect. 4.5.3), and it does not explicitly define an immobile region (a) or a transitional area (b); however, the output is very similar to the typical deformation shown in Fig. 4.18 and demonstrates the existence of an ellipse-like quasi un-deformable region.

Novikov and Ushmaev (2005) and Si et al. (2015) studied the principal deformation of fingerprints, that is, the typical way a fingerprint distorts. The analysis is performed by reducing (through PCA) the dimensionality of the displacement vectors obtained by registering the elastic distortion in fingerprint pairs. With fingerprints being pre-aligned with respect to rigid transformations, the main eigenvectors just encode the principal modes of variation of the distortion with respect to a neutral print. Evaluations performed by Novikov and Ushmaev (2005) over FVC2002 databases reveal that torsion and traction along the two axes are among the few principal distortions of fingerprints; the experiments also confirm the presence of a quasi un-deformable region. A new distorted fingerprint database (called Tsinghua DF database) was collected by Si et al. (2015) to learn the principal components of distortion (Fig. 4.19).

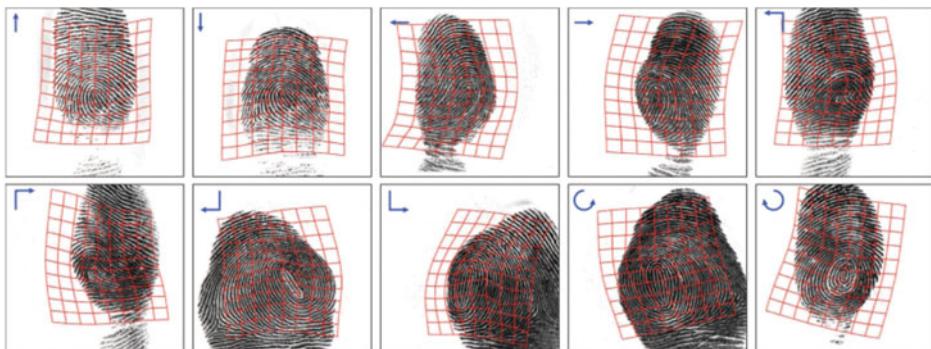


Fig. 4.19 Examples of 10 distortion types in Tsinghua DF database. The blue arrows represent the directions of force or torque, and red grids represent the distortion grids which are calculated from matched minutiae between the normal fingerprint and the distorted fingerprint. © IEEE. Reprinted, with permission, from Si et al. (2015)

4.5.2 Tolerance Box Adaptation

Distortion is dealt with by relaxing the spatial relationships between minutiae: in particular, parameter values r_0 and θ_0 in Eqs. (4.6) and (4.7) are increased, thus making the pairing constraints less stringent.

In global minutiae matching, distortion may significantly alter the relative distance of two minutiae far away from the principal pair (i.e., the pair used for registration) because of the spatial integration of the skin stretching/compression, and therefore large tolerance boxes need to be used. As a consequence, the probability of falsely matching fingerprints from different fingers increases. In local minutia matching the tolerance box technique is usually more effective; however, particular care must be taken in the consolidation phase to avoid diminishing this advantage.

Tolerance boxes are defined in polar coordinates (sectors of circular annuluses) in the approaches by Jain et al. (1997) and Luo et al. (2000), where edit distance is used for matching pre-aligned minutiae. The size of the tolerance boxes is incrementally increased moving from the center towards the borders of the fingerprint area in order to compensate for the effect of distortion. Jea and Govindaraju (2005) and Chen et al. (2006b) suggest that, unlike the radial tolerance, the angular tolerance must decrease with the distance from the centre. Zheng et al. (2007) argue that a larger tolerance box can be used to match the subsets of ARM (Absolutely Reliable Minutiae) in \mathbf{T} and \mathbf{I} , since the risk of matching by chance markedly reduces.

Hao et al. (2002) method not only increases the tolerance with the radius but also implements an iterative error propagation algorithm, where tolerance boxes of the unmatched minutiae are iteratively adapted according to the actual positional errors of the spatially close (already) matched minutiae. A similar idea is exploited by Tong et al. (2008), who define the Relative Location Error (RLE) between two corresponding minutiae according to the location error of neighbouring minutiae.

Finally, Lee et al. (2002) and Bhowmick and Bhattacharya (2004), instead of relaxing minutiae distance constraints, proposed to normalize the minutiae distance according to the local ridge frequency. Suppose that a portion of a fingerprint is distorted by a traction force that increases the distance between two minutiae. Then the local ridge frequency decreases accordingly and therefore a simple normalization of the distance by the frequency results in a sort of distortion-tolerant distance.

4.5.3 Warping

These techniques (a.k.a. non-rigid registration) explicitly address the fingerprint distortion problem, by allowing one of the two fingerprints to be locally warped to maximize the number of matching minutiae. In general, these methods start from a solution determined

under the rigid-matching constraint that is relaxed in the second stage: this allows to deal with the large number of degrees of freedom and to improve efficiency.

Bazen and Gerez (2003) attempt to find a smoothed mapping between the template and the input minutiae feature set. Minutiae pairing is initially computed through a local approach and a consolidation step; then the size of the tolerance boxes is reduced, and a thin-plate spline (TPS) model is used to deal with the non-linear distortions. Through an iterative procedure, which starts from the initial pairing, the minutiae in the input fingerprint are locally moved (according to the model smoothness constraints) to best fit the template minutiae. The authors report that they obtained a significant improvement when the distortion-tolerant matcher was used instead of the rigid matcher. Some researchers have suggested replacing TPS with other warping techniques:

- Novikov and Ushmaev (2004) technique relies on the elasticity theory in solid state mechanics and defines the elastic distortion as a solution of Navier linear Partial Differential Equations (PDE).
- Liang and Asano (2006) used multi-quadratic base functions in conjunction with Radial Basis Function (RBF) interpolation. This allows to better control the range of influence of each control point (i.e., minutia pair); a consequence is that the distortion correction is softer where control points are widely spaced and stronger where they are closer together. Another quadratic model controlled by 12 parameters and including a penalty term to favor smooth transformations was proposed by Cao et al. (2009).
- Meenen et al. (2006) warping is implemented through a two-dimensional Taylor series expansion truncated at the second degree.

Ross et al. (2005, 2006) proposed two techniques for computing the *average deformation model* of a fingerprint impression by using the TPS warping technique. Given several impressions of a finger, they estimate the average deformation of each impression by comparing it with the rest of the impressions of that finger. The average deformation is developed using TPS warping and is based on minutiae correspondences in Ross et al. (2005) and on ridge curve correspondences in Ross et al. (2006). The estimated average deformation is then utilized to pre-distort the minutiae points in \mathbf{T} image before matching with \mathbf{I} . Experimental results show that the use of an average deformation model leads to a better alignment between minutiae resulting in higher matching accuracy. Ross et al. (2006) argue that modeling the distortion using ridge curve correspondences offers several advantages over minutiae correspondences, resulting in improved matching performance (see Fig. 4.20). Unlike minutiae points, which can be sparsely distributed in certain regions of a fingerprint image, ridge curves are present all over the image domain, thereby permitting a more reliable estimate of the distortion.

Almansa and Cohen (2000) introduced a 2D warping algorithm controlled by an energy function that has to be minimized in order to find the optimal mapping. The first term of the energy function requires the two minutiae sets to spatially coincide, whereas the

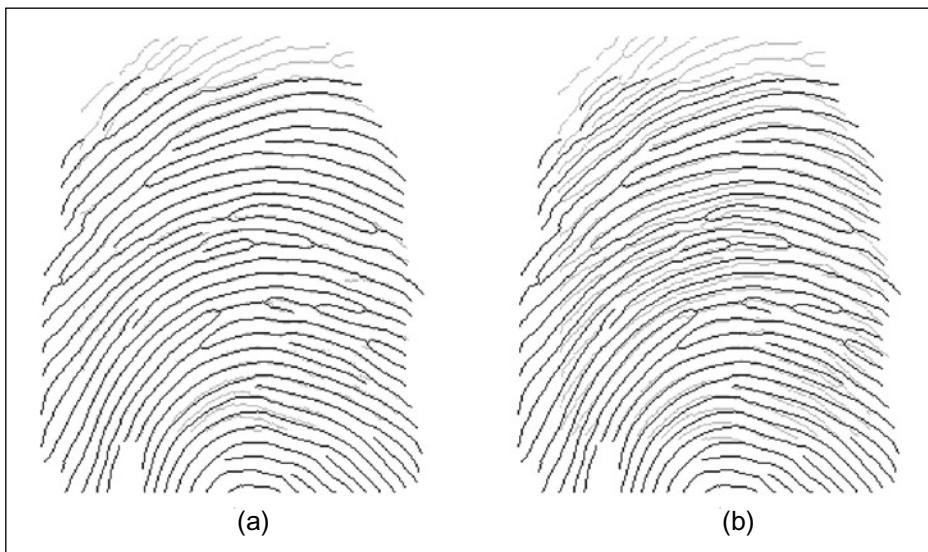


Fig. 4.20 **a** Alignment of two fingerprint images using the ridge curve correspondences proposed by Ross et al. (2006); **b** The same fingerprints are aligned using minutiae minutiae correspondences. Both use the TPS warping model. © IEEE. Reprinted, with permission, from Ross et al. (2006).

second term introduces a penalty that increases with the irregularity of the warping. Unfortunately, energy minimization is performed with a two-step iterative algorithm whose convergence may be critical and whose time complexity is high. A computationally more efficient energy minimization technique was introduced by Kwon et al. (2007) whose distortion correction is based on a triangular mesh model and a gradient-based optimization technique. Based on their empirical results, Kwon et al. (2007) conclude that this approach is more robust than TPS in the presence of outliers (i.e., false minutiae or wrong minutiae correspondence).

4.5.4 Dense Registration

Dense registration builds a pixel-level correspondence between two fingerprints of the same finger. Such a precise alignment can be beneficial for fingerprint mosaicking (Sect. 2.9) and to improve matching accuracy in presence of distortion. Three dense fingerprint registration approaches have been introduced:

- the method proposed by Si et al. (2017) is based on block-based image correlation and global energy minimization (see Fig. 4.4).

- Cui et al. (2018) makes use of 2D-Phase demodulation to compute dense displacement field for alignment. The input image is treated as the modulated signal, the reference image as the carrier signal, and the target of image registration is to recover the distortion field as the message signal.
- a Convolutional Neural Network (with Siamese architecture) is trained to regress pixel-wise displacements in the method by Cui et al. (2019, 2021). The ground-truth data for training are automatically generated by previous techniques.

The third method was shown to be more robust in case of highly distorted and poor quality fingerprints (some examples are reported in Fig. 4.21). Dense registration techniques usually require an initial alignment (e.g., TPS-based registration at minutiae level) and can be computationally expensive.

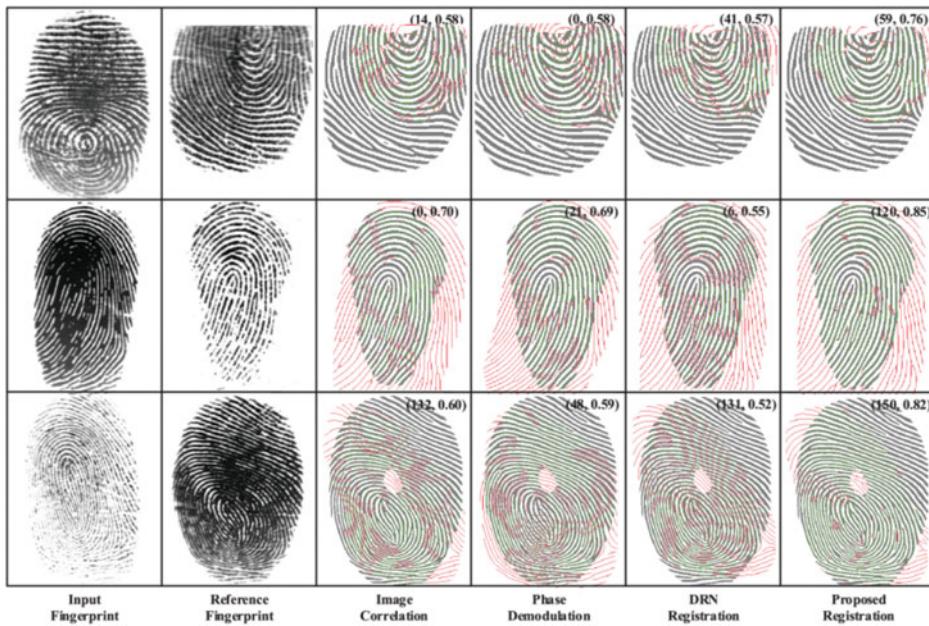


Fig. 4.21 Dense registration of 3 fingerprint pairs according to Si et al. (2017) (3rd column), Cui et al. (2018) (4th column), Cui et al. (2019) (5th column) and Cui et al. (2021) (6th column). A larger “green coverage” denotes better registration. © IEEE. Reprinted, with permission, from Cui et al. (2021)

4.5.5 A-Priori Distortion Removal

The idea consists of a-priori correction for distortion from fingerprint samples (a.k.a. distortion *rectification*) so that a conventional matcher using rigid alignment can work properly even on distorted fingerprints. Such distortion rectification can be performed during enrollment to: (i) speed up successive matching stages and (ii) discard samples heavily affected by distortion.

The most evident effect of distortion is the local compression/stretching of the fingerprint ridges and valleys. To eliminate the undesired effects of distortion, Senior and Bolle (2001) proposed normalizing the fingerprint image to a *canonical* form by deriving a fingerprint representation where all the ridges are equally spaced. The fingerprint images are binarized and thinned, and ridge lines are approximated by spline curves. The average inter-ridge distance is then determined, and two ridge dilatation maps are computed that encode the local deviation of the inter-ridge distance from the average one. Transformation into a canonical form is then obtained by integration according to the two dilatation maps (Fig. 4.22) while such a normalization can effectively compensate for cross-ridge stretching/compression, it cannot handle along-ridge distortion component very well because ridge spacing is not useful for this purpose. Furthermore, the ridge period is not completely uniform either in the same finger nor in different fingers; hence, normalizing inter-ridge distance may lose discrimination information. Finally, while canonical representations can be adequate for proprietary solutions, they conflict with efforts in establishing standard and interoperable fingerprint templates.



Fig. 4.22 Transformation of a fingerprint skeleton (left) into canonical form (right) (Senior & Bolle, 2001). © IEICE. Reprinted, with permission, from Senior and Bolle (2001)

Si et al. (2015) proposed a learning based distortion rectification. If a given input fingerprint is detected as distorted, distortion rectification is performed to transform the input fingerprint to canonical form, otherwise the fingerprint image is left untouched to avoid potentially disturbing transformations. Through the observation of a large number of fingerprints, Si et al. (2015) found that distorted fingerprints are quite different in terms of local orientation and frequencies compared with normal fingerprints. Hence their distortion detector is implemented as a binary SVM classifier trained on pose-aligned orientation and frequency maps. The distortion rectification algorithm (Fig. 4.23) consists of two stages: the offline stage and the online stage. In the offline stage, a reference distorted fingerprint database is constructed by applying different distortion field to the normal fingerprints. Here, the distortion field is sampled from the statistical model of the real distortion field. In the online stage, given an input distorted fingerprint, its nearest neighbor is found in the reference database, and the corresponding distortion field is used to transform the input fingerprint into a normal one. Si et al.'s (2015) approach was evaluated on different databases; results show that: (1) on the databases containing many distorted fingerprints (FVC2004 DB1 and TSINGUA DF database), the proposed algorithm significantly improves the matching accuracy; (3) on a database without severely distorted fingerprints (FVC2006 DB2_A), the proposed algorithm has no negative impact.

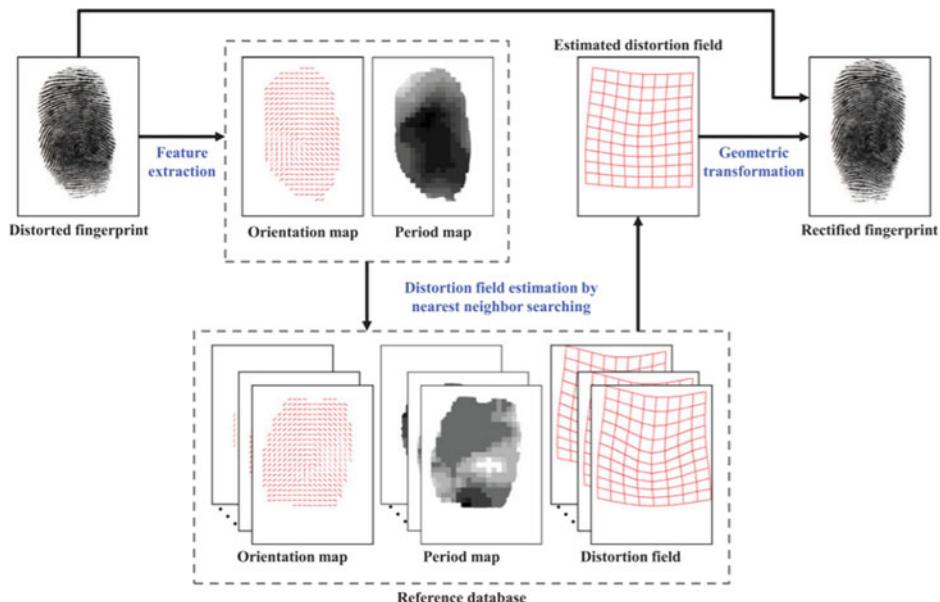


Fig. 4.23 Learning-based distortion rectification in the approach by Si et al. (2015). © IEEE. Reprinted, with permission, from Si et al. (2015)

Gu et al. (2017) introduced another learning based approach to increase efficiency and accuracy of Si et al. (2015) method. The main differences are as follows:

- an improved pose estimation algorithm, where the fingerprint center is first determined by a Hough forest based approach and the finger direction is then estimated by a support vector regression method.
- the fingerprint distortion field is estimated by an SVM regressor taking in input both the orientation and frequency maps, and providing as output the coefficients of the principal components of the distortion field.

The above two methods estimate distortion parameters based on the ridge frequency map and orientation map of fingerprints, which may be unreliable due to low image quality. To alleviate this, Dabouei et al. (2018) developed a rectification model based on a deep Convolutional Neural Network to regress distortion parameters directly from the input image. Pose estimation and distortion detection are not required since the fingerprint image is simply centered according to the center of mass of the foreground area.

4.6 Feature-Based Matching Techniques

Some of the reasons that induce designers of fingerprint recognition techniques to search for additional fingerprint distinguishing features, beyond minutiae are listed below:

- Additional features may be used in conjunction with minutiae (and not as an alternative) to increase system accuracy and robustness. It is worth noting that several feature-based techniques use minutiae for pre-alignment or to define anchor points.
- Reliably extracting minutiae from poor quality fingerprints is difficult. Although minutiae may carry most of the fingerprint discriminatory information, they do not always constitute the best tradeoff between accuracy and robustness for poor quality fingerprints.
- Non-minutiae-based methods may perform better than minutiae-based methods when the surface area of fingerprint sensor is small. In fingerprints with small area, only 4–5 minutiae may exist and in that case minutiae-based algorithm do not perform satisfactorily.
- The variability in the number of minutiae and the lack of a natural ordering, make it difficult to derive a robust fixed-length minutiae-based representation. Textural features can be better suited to this purpose.

The most commonly used non-minutiae features are as follows:

1. Local orientation and frequencies

2. Geometrical attributes and spatial relationship of the ridge lines
3. Handcrafted (general-purpose) textural features
4. Deep network learned features
5. Level 3 features (e.g., sweat pores).

Analogous to minutiae matching, texture analysis can be performed at both global or local level:

- Global texture analysis fuses contributions from different characteristic regions into a global measurement. The main advantage of global methods is that they provide fixed-length feature vectors that can be matched very efficiently (also in conjunction with cryptographic methods). On the other hand, in global methods: (i) some spatial information may be lost, (ii) in case of small overlap between two fingerprints in comparison, the representation of the two fingerprints can be quite different.
- Local texture analysis is usually more effective than global feature analysis. We already pointed out in Sect. 4.4.2 the relevance of texture-based local structures in the evolution of local structure matching. Unfortunately, working at local level often results in variable length representations that require non-trivial matching approaches.

Designing methods working on the raw fingerprint images (with no need of priori pose estimation) while providing robust fixed-length representations is still an open issue that recent deep learning techniques are trying to address.

4.6.1 Early Global Methods

Coetze and Botha (1993) and Willis and Myers (2001) proposed analyzing fingerprint texture in the Fourier domain. Although ridges in the spatial domain transform to a fairly constant frequency (in the frequency domain), the distinguishing characteristics of a fingerprint such as the specific ridge orientation and the minutiae manifest themselves as small deviations from the dominant spatial frequency of the ridges. A “wedge-ring detector” is then used to perform the analysis in the frequency domain; the harmonics in each of the individual regions of the detector are accumulated, resulting in a fixed-length feature vector that is translation, rotation, and scale invariant.

The most popular technique to match fingerprints based on texture information remains the FingerCode approach by Jain et al. (2000). The fingerprint area of interest is tessellated with respect to the core point (see Fig. 4.24).

The texture information in each sector is decomposed into separate channels by using a Gabor filterbank (ref. Sect. 3.6.2). In their experimentation, Jain et al. (2000) obtained good results by tessellating the area of interest into 80 cells (five bands and 16 sectors), and by using a bank of eight Gabor filters (eight orientations, and one fixed scale

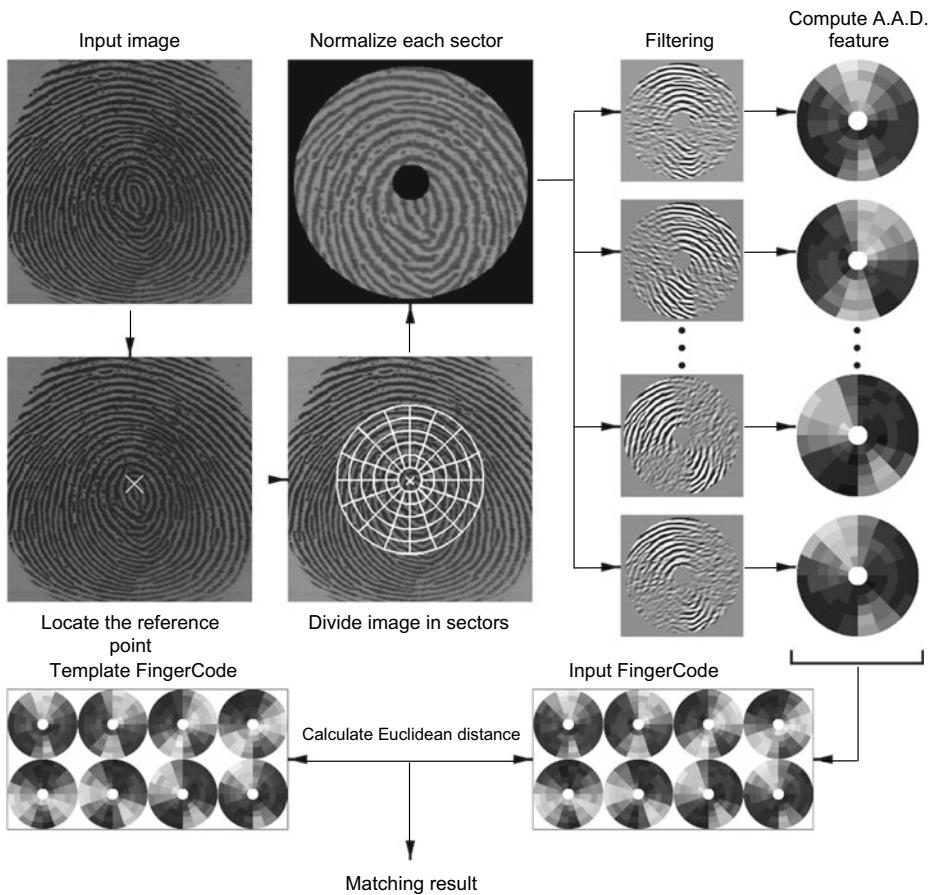


Fig. 4.24 Diagram of Jain et al.’s (2000) FingerCode approach. © IEEE. Reprinted, with permission, from Jain et al. (2000)

= 1/10 for 500 dpi fingerprint images). Therefore, each fingerprint is represented by a $80 \times 8 = 640$ fixed-size feature vector, called the *FingerCode*. Matching two fingerprints is then translated into matching their respective FingerCodes, which is simply performed by computing the Euclidean distance between two FingerCodes.

Several variants of the Fingercode approach were introduced to: (i) make it less sensitive to pose estimation and more robust for small-area sensors (Jain et al., 2001; Ross et al., 2002, 2003; Sha et al., 2005); (ii) adding further information to the cells (Sha et al., 2003); (iii) turn it into a localized version (Benhammadi et al., 2007).

4.6.2 Local Orientation and Frequencies

We know that most of the local texture information is contained in the orientation and frequency images. Several methods have been proposed where a similarity score is derived from the correlation between the aligned orientation images of the two fingerprints: Qi and Wang (2005), Cheng et al. (2004), Yager and Amin (2004), Kulkarni et al. (2006), Zhang et al. (2007), and Zheng et al. (2007). The alignment can be based on the orientation image alone (see Sect. 4.3.7) or delegated to a minutiae matching stage. Gu et al. (2006) also used orientation image in their hybrid matching technique, but unlike other researchers they used a model-based approximation of the orientation image (see Sect. 3.2.6). Their experimental results show that not only this allows to reduce the template size (in fact only the model parameters have to be stored) but also improve accuracy due to the robustness of the model to local perturbation of the orientation image. Wan and Zhou (2006) used frequency image correlation in conjunction with minutiae matching to produce a final score. Since the frequency images directly computed from the fingerprints may lack in robustness against noise and distortion, a polynomial model is proposed to approximate the coarse frequency map, thus making the resulting features more robust.

4.6.3 Geometrical Attributes and Spatial Relationship of the Ridge Lines

The use of spatial relationship among ridges forms the basis of the earlier methods proposed by Moayer and Fu (1986) and Isenor and Zaky (1986). In the former, tree grammars are introduced to classify ridge line patterns after that they are binarized and thinned. In the latter, incremental graph matching was carried out to compare a set of ridges arranged in graph structures (ridges are the graph nodes and arches are defined according to ridge adjacency and visibility). Other researchers have focused on techniques that rely, at least partially, on ridge matching. The ridges representation is usually obtained by sampling points (at fixed intervals) along each thinned ridge.

In Chen et al. (2006b) ridge information is associated with the minutiae they originate from; once the minutiae are paired with a local structure-based algorithm, the computation of the final score is consolidated by taking into account the spatial matching of the ridge points; experimental results show that the contribution due to ridge sampling is significant.

Xie et al. (2006a), Feng et al. (2006), and Feng and Cai (2006) explicitly exploit ridge relationships: for each sampled point they annotate the labels of the two neighboring ridges. More details are given below:

- In Xie et al. (2006) ridge-matching is performed by comparing neighboring information for all the pre-matched pairs of ridges: two ridges (one from **T** and one from **I**) are pre-matched if they have similar length and curvature.

- In Feng et al. (2006), the ridge and minutiae matching is performed incrementally, starting from a local structure-based alignment.
- Feng and Cai (2006) encode the geometric relationships among points sampled over ridges based on RCS (Ridge Coordinate System); an RCS is defined for every ridge R by using a minutia to set the origin and the direction of the reference axes and the ridge count as a metric for measuring distances. The RCS coordinates of the points of all the ridges (except R) form the feature vector associated to R . These feature vectors, which are invariant under translation and rotation, are used for alignment and matching through a greedy algorithm.

Zhang et al. (2007) overcome the problems introduced by the ambiguity between ridge ending and bifurcation minutiae by sampling both the ridge and valley on the two sides of each minutia. Feng et al. (2006) simplify the successive ridge matching, by implementing a post-processing stage where the ridge pattern is regularized by: (i) disconnecting closed ridges (i.e., loops), (ii) splitting into three parts the ridges associated with bifurcations, and (iii) removing short ridges. In the method by Choi et al. (2011) ridge features are composed of four elements: ridge count, ridge length, ridge curvature direction, and ridge type. Finally, edge corners on fingerprint ridges are considered in the approach by Nachar et al. (2020).

In conclusion, we observe that ridge matching methods typically require good quality ridge maps. Dense registration methods in recent years (Sect. 4.5.4) share some similarity with ridge matching, but are more flexible in dealing with noise and distortion.

4.6.4 Handcrafted Textural Features

Some authors demonstrated that the use of general-purpose textural features can be advantageous for fingerprint recognition. In particular, matching techniques based on key points and local descriptors such as SIFT, SURF, KAZE (Szeliski, 2011; Tareen & Saleem, 2018) have been shown to be effective to match fingerprint with small overlap and limited number of minutiae.

- *Wavelets*: Tico et al. (2001) suggested using wavelet domain features and claimed that their method achieved performance comparable to the Gabor-based one, but had the advantage of avoiding any pre-processing such as core point detection. Huang and Aviyente (2004a, b) focused on choosing the best wavelet basis for fingerprint matching. Multi-resolution approaches based on the Discrete Wavelet Transform (DWT) were also introduced by Chebira et al. (2007), Nanni and Lumini (2007) and Bouchafra and Amira (2008). Rowe (2007) used the coefficients of a wavelet decomposition to match the large amount of textural data provided by a multispectral fingerprint scanner. Filterbanks and wavelets are combined in the work by Li et al. (2012), and rotated

wavelet filters were used by Bharkad and Kokare (2012). Amornraksa and Tachaphet-piboon (2006) report that in their experiments that Discrete Cosine Transform (DCT) coefficients provided better accuracy with respect to DWT coefficients.

- *GLCM*: the use of Gray-level co-occurrence matrix (GLCM) was proposed by Khalil et al. (2009): the GLCM are computed at different angles and their information are then summarized by compact measures (correlation, contrast, energy, and homogeneity).
- *LBP*: Nanni and Lumini (2008), after a minutiae-based alignment, decompose the image into several overlapping blocks, and for each block: (i) apply a bank of Gabor filters, (ii) extract LBP (Local Binary Pattern) features; (iii) use Euclidean distance to compare LBP histograms. LBP features (Zhang et al., 2004) are grayscale invariant statistics that can be computed as the difference between the gray value of a central pixel and the average gray value over its circular neighborhood.
- *SIFT*: Park et al. (2008) first adopted Scale Invariant Feature Transformation (SIFT) for fingerprint recognition. SIFT extracts repeatable characteristic feature points from an image and generates descriptors representing the texture around the feature points. SIFT has also been used by Malathi and Meena (2011), Yamazaki et al. (2015), and Aravindan and Anzar (2017).
- *KAZE*: Mathur et al. (2016) argued that SIFT and SURF features are subject to isotropic blurring which KAZE multiscale descriptors can alleviate. Therefore they propose an Accelerated-KAZE implementation which performs better than minutiae and SIFT matching in the partial-to-partial matching scenario characterizing fingerprint recognition on smartphones.

4.6.5 Deep Features

The first neural network-based approach for fingerprint recognition date back to a few decades (Sjogaard, 1992; Baldi & Chauvin, 1993; Quek et al., 2001; Coetzee & Botha, 1990); Melin et al., 2005). Only in the last few years (with the deep learning revolution) some effective solutions have emerged.

The DeepPrint approach by Engelsma et al. (2021) incorporate fingerprint domain knowledge, including alignment and minutiae detection, into a deep network architecture to maximize the discriminative power of its representation. A diagram of this approach is shown in Fig. 4.25. The input fingerprint is aligned by a Localization Network (no reference points are needed for alignment) and passed to a Base-Network which is followed by two branches; (i) the first branch extracts a 96-dimensional texture-based representation; (ii) the second branch extracts an 96-dimensional minutiae-based representation, guided by a side-task of minutiae detection (via a minutiae map which does not have to be extracted during inference). The texture-based and minutiae-based representations are concatenated into a 192-dimensional code (768 bytes) which is further compressed

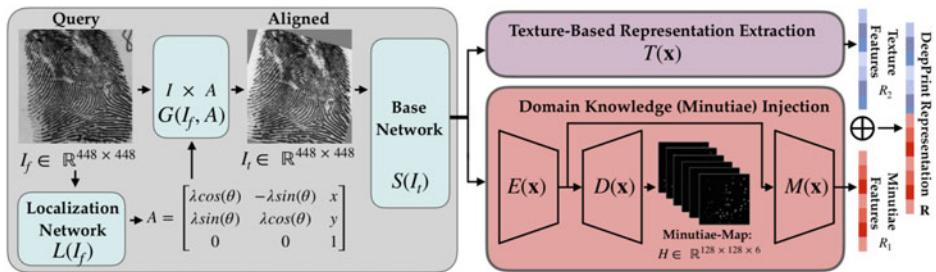


Fig. 4.25 Functional diagram of the DeepPrint approach by Engelsma et al. (2021). © IEEE. Reprinted, with permission, from Engelsma et al. (2021)

to 200 byte fixed-length representation. Matching two fingerprints can be then translated in computing the cosine similarity between their fixed-length representations. The authors showed that this approach is competitive with state-of-art fingerprint recognition approaches, especially for rolled and highly distorted fingerprints.

Lin and Kumar (2019) introduced a multi-Siamese CNN to compare contactless with contact-based fingerprints. Representation learning can be very useful in this case because of the very different textural appearance of the two types of fingerprints. The authors first train the model using fingerprint minutiae, ridge map, and specific regions of ridge map. The network is then used to generate a deep fingerprint representation using a distance-aware loss function.

Zhang et al. (2019) specifically addressed fingerprint recognition on mobile devices where the limited area of fingerprint sensors brings challenges of partial fingerprint matching. Their model employs a couple of deep convolutional neural networks to learn both high-level global features and low-level minutia features, by using triplet loss. Score level fusion is then performed to combine global similarity and spectral correspondence of minutiae matching.

4.6.6 Pore Matching

Arrangement of sweat pores along fingerprint ridges is undoubtedly highly discriminant but, as pointed out by several researchers, reliable detection of sweat pores requires high-resolution scanners and is very challenging in low-quality fingerprints. Furthermore, in some research works, the relative contribution of pores might have been overestimated because the baseline matcher used was not state-of-the-art. Pore detection techniques have been already discussed in Sect. 3.10; hereafter we focus on pore matching.

In the early approach by Stosz and Alyea (1994) pore matching is performed (in a human-assisted or semi-automatic mode) by: (i) subdividing the images in small regions

and retaining only discriminant regions; (ii) pre-aligning the regions through intensity correlation; (iii) counting the numbers of spatially coincident pores for each pair of regions. Based on the above algorithm, Roddy and Stosz (1997) later conducted a statistical analysis of pores and presented a model to predict the upper bound on performance of a pore-based automated fingerprint system. For example, they demonstrated that the probability of occurrence of a particular combination of 20 ridge-independent pores is 5.186×10^{-8} .

Kryszczuk et al. (2004) studied the distinctiveness of minutiae and pore features in fragmentary fingerprint comparison by using images acquired at about 2,000 dpi. For this purpose, they also implemented a skeletonization-based pore detection algorithm with more anatomical constraints with respect to Stosz and Alyea (1994) algorithm. Kryszczuk et al. (2004) concluded that the use of pores can offer at least a comparable recognition potential from a small area fingerprint fragment, as minutiae features offer for fragments of larger area.

Jain et al. (2007) conducted a systematic study to determine how much performance gain one can achieve by introducing Level 3 features in AFIS. They noted that skeletonization is effective for pore extraction only when the image quality is very good and the image resolution is very high. Level 1 (orientation images), Level 2 (minutiae), and Level 3 (pores and ridge contours) features are then hierarchically matched, leading to early rejection in case of very low similarity. Pores and ridge contours are locally matched (in the neighbourhoods of the already paired minutiae) using the Iterative Closest Point (ICP) algorithm. Jain et al. (2007) evaluated their approach over a database acquired with a 1,000 dpi commercial scanner and showed that Level 3 features carry significant discriminatory information. There is a relative reduction of 20% in the EER of the matching system when Level 3 features were employed in combination with Level 1 and 2 features. The performance gain is consistently observed across various quality fingerprint images.

Chen and Jain (2007) proposed an algorithm, based on local phase symmetry, to extract other Level 3 features such as dots and incipient ridges. According to some latent examiners these features are often more valuable and reproducible than pores for the purpose of partial latent fingerprint matching. Dots and incipients were added to the minutiae set and matched as they were normal minutiae (the origin of an incipient ridge is placed in its midpoint). This led to a significant accuracy improvement when matching partial fingerprints.

In the work by Abhyankar and Schuckers (2010) pores are encoded by graphs via Delaunay triangulation. An advantage of this representation is that pore detection inaccuracies can only locally affect the triangulation procedures. Matching is then performed by Deng and Huo (2005) method relying on nearest-neighbour based structures. Other graph-matching based approaches were proposed by Xu et al. (2019a, b).

Since pores (and pores neighbours) are more similar to each other with respect to minutiae (and minutiae-neighbours), establishing a coarse pairwise pore correspondences

can be critical. The methods introduced by Liu et al. (2010b, 2011, 2020) maintain one-to-many correspondences and use weighted RANSAC for robust alignment.

Local pore descriptors have been shown to be useful to determine initial pore correspondences:

- Zhao et al. (2010) proposed a pore–valley descriptor (PVD) to characterize pores based on their location and orientations and also considering local ridge orientations and valley structures. A coarse-to-fine matching strategy is introduced to determine pore correspondences.
- The descriptors proposed by Xu et al. (2019b) are built by considering edges connecting pairs of pores. The feature vector associated to each edge consists of the edge distance and some relative angles between the edge orientation and the pore orientations.
- Liu et al. (2020) trained a CNN to learn deep features for each pore. The obtained (128-dimensional) descriptors can be simply compared by Euclidean distance metric. This helps to initialize one-to-many correspondences which are further refined with

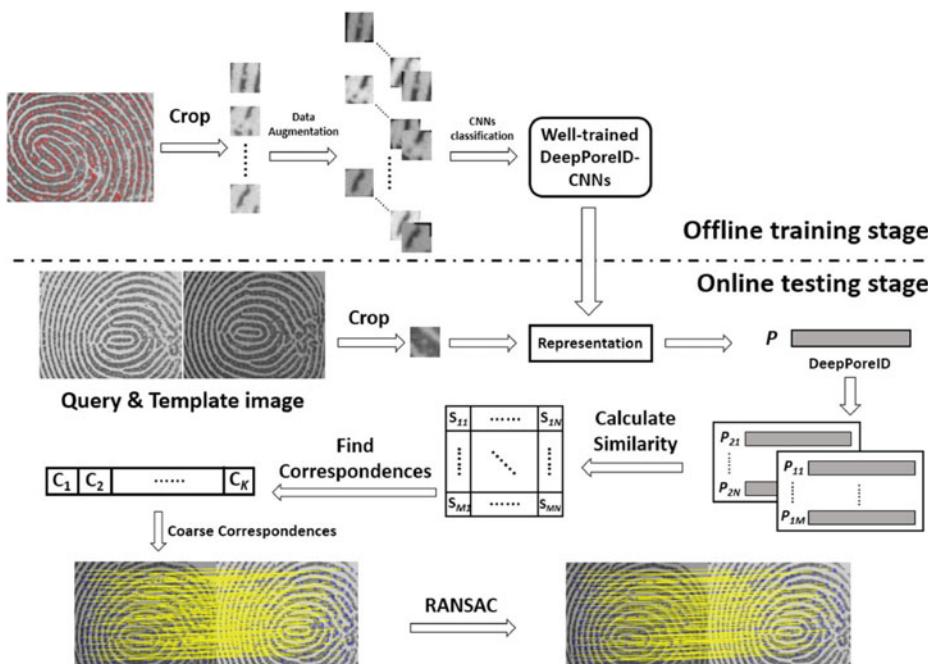


Fig. 4.26 An overview of the pore matching approach by Liu et al. (2020). © Elsevier. Reprinted, with permission, from Liu et al. (2020)

weighted RANSAC approach (see Fig. 4.26). State-of-the-art accuracy was reported by the authors on two high resolution fingerprint datasets (PolyU HRF).

4.7 Comparing the Performance of Matching Algorithms

Within the biometric recognition context, a benchmark is defined by a database and an associated testing protocol. The protocol defines the subsets of images that can be used for training and testing, the pair of images that have to be compared, the performance metrics to be used and how they must be computed. Collecting fingerprint databases is expensive and prone to human errors. Chapter 7 addresses this problem in more details and introduces synthetic fingerprint generation as a viable solution for fingerprint algorithm testing. Using exactly the same benchmark is essential to compare the performance of different algorithms.

4.7.1 Fingerprint Databases

Before the organization of the first fingerprint verification competitions FVC2000, the only large public domain fingerprint datasets were the National Institute of Standards and Technology (NIST) databases:

- NIST DB 4 (Watson & Wilson, 1992a), NIST DB 9 (Watson & Wilson, 1992b), NIST DB 10 (Watson, 1993), and NIST DB 14 (Watson, 1993) contain thousands of images scanned from rolled inked impressions on cards, which are quite different from live-scan dab images.
- NIST DB 24 (Watson, 1998) contains 100 live sequences (capture video) from 10 individuals. In principle, static frames could be extracted from the videos and used for performance evaluation; on the other hand, most of the videos have been taken under particular conditions to study the effect of finger rotation and plastic distortion and therefore are not well suited for overall system evaluations.
- NIST DB 27 (Garris & McCabe, 2000) was released to test the performance of latent fingerprint identification. Latent fingerprint images of varying quality were provided together with their corresponding rolled impressions taken from fingerprint cards. Minutiae data manually extracted by human experts were also provided. NIST DB 27 is a valuable source of information for studying the difficult problems of latent fingerprint enhancement and matching.

Unfortunately NIST fingerprint databases have been withdrawn in 2018 and are no longer available for purchase or download. Furthermore, although the above databases constitute

an excellent benchmark for AFIS development, they are not well suited for the evaluation of algorithms operating on live-scan (dab) images.

FVC campaigns (2000, 2002, 2004, and 2006) were organized with the aim of providing fingerprint databases to any interested researcher and to track performance of the state-of-the-art fingerprint matching algorithms. Fortunately, most of the authors now report the results of their experiments on one or more of these databases according to the proposed protocol, thus producing results that can be compared across the whole scientific community. It is hoped that this will become a common practice for all the scientists and practitioners in the field. For this purpose, we have included the complete FVC2000, FVC2002, and FVC2004 databases in the DVD accompanying this book. FVC2006 can be obtained upon request following the instruction in BioLab (2007). Readers are invited to consult the FVC reports (also included in the DVD) where they will find all the details necessary to set up a test session identical to that performed in the competitions. Table 4.1 provides a brief summary of FVC databases. The difficulty of each database (as reported in the note field) depends on several factors: the population, the perturbations voluntarily introduced, the quality of the scanner, etc.; the difficulty reported in Table 4.1 derives from an a-posteriori analysis of the results achieved by top performing participants (see also Table 4.3). In general, FVC databases should not be used to infer or compare the scanner quality; in fact, except for FVC2006, the protocol and volunteers were not the same for the different scanners; furthermore, other indicators such as the Failure To Enroll (FTE) should be taken into account to perform a fair scanner comparison. We recommend scientists and practitioners to test their systems according to the scenarios reported in Table 4.2.

Besides NIST and FVC databases, other fingerprint collections can be used for testing fingerprint algorithms.

- TSINGUA DF³: this database was collected for studying the problem of distorted fingerprints, including distortion detection, distortion rectification, and dense registration (Si et al., 2015). It contains 320 pairs of normal fingerprints and distorted fingerprints from 185 different fingers.
- PolyU HRF: consists of two high-resolution (1200 dpi) fingerprint databases provided by the Hong Kong Polytechnic University (PolyU). Each of the database includes 1,480 images from 148 fingers. For each finger, 10 samples were collected in two sessions. Because of the high resolution it was used to study pore extraction and matching. Unfortunately, this dataset is no longer available to the research community.
- CASIA v5.0⁴: contains 20,000 fingerprint images of 500 subjects. Fingerprints were captured using URU4000 fingerprint sensor in one session. Each volunteer contributed 40 fingerprint images of 8 fingers, 5 samples per finger.

³ <http://ivg.au.tsinghua.edu.cn/dataset/TDFD.php>.

⁴ http://english.ia.cas.cn/db/201611/t20161101_169922.html.

Table 4.1 A summary of FVC databases. In the database size column, the notation MxN, denotes M fingers and N samples per finger. In all the competitions the first three databases were acquired by selecting commercial scanners of different types, including large-area optical, small area optical, solid state capacitive and thermal sweep. The fourth database was synthetically generated by using the SFinGe tool (see Chap. 7).

Competition	Number of databases	Size of each database: A—Evaluation set B—Trainin set	Notes
FVC2000 Maio et al. (2002a)	4	A: 100×8 B: 10×8	<ul style="list-style-type: none"> – Volunteers are mainly unhabituated students – Two sessions, no quality check – Low/Medium difficulty (DB1, DB2, DB4); – Medium/High difficulty (DB3)
FVC2002 Maio et al. (2002b)	4	A: 100×8 B: 10×8	<ul style="list-style-type: none"> – Volunteers are mainly unhabituated students – Three sessions, no quality check – Voluntarily exaggerated perturbations: displacement, rotation, wetness and dryness – Low difficulty (DB1, DB2, DB3, DB4)
FVC2004 Maio et al. (2004) Cappelli et al. (2006)	4	A: 100×8 B: 10×8	<ul style="list-style-type: none"> – Volunteers are mainly unhabituated students – Three sessions, no quality check – Voluntarily exaggerated perturbations: distortion, wetness and dryness – Medium difficulty (DB1, DB2, DB3, DB4)
FVC2006 BioLab (2007)	4	A: 140×12 B: 10×12	<ul style="list-style-type: none"> – Heterogeneous population also includes unhabituated manual workers and elderly people. No quality check – The final datasets were selected from a larger database by choosing the most difficult fingerprints according to a quality index – High difficulty (DB1), Medium difficulty (DB3), Low difficulty (DB2, DB4)

Table 4.2 Recommended benchmarks for the evaluation of algorithms in different contexts: performance should be reported on the largest subset of databases listed for the scenario of interest.

Scenario	Benchmark
Algorithms optimized to work with large-area touch sensors (e.g., AFIS, government applications)	FVC2000—DB3 FVC2002—DB1, DB2 FVC2004—DB1 FVC2006—DB2
Algorithms designed to work with large-, medium- and small-area touch sensors	FVC2000—DB1, DB2, DB3, DB4 FVC2002—DB1, DB2, DB3, DB4 FVC2004—DB1, DB2, DB4 FVC2006—DB1, DB2, DB4
Algorithms to be used with sweep sensors	FVC2004—DB3 FVC2006—DB3
Test aimed at showing the algorithms robustness with respect to plastic distortion	FVC2004—DB1, DB2, DB3, DB4

- MOLF (2015): the Multisensor Optical and Latent Fingerprint (Sankaran et al., 2015) contains 19,200 multisensor, multi-spectral, flat and slap fingerprint images of 1000 fingers (100 subjects) obtained from three different sensors along with mated latent fingerprints. Moreover, the latent fingerprints have manually annotated features. This dataset is useful to study sensor interoperability and latent-to-livescan fingerprint recognition.
- FingerPass (2012): this database was introduced by Jia et al. (2012) to study fingerprint sensor interoperability. It consists of almost 80,000 fingerprint images from 90 subjects on nine different fingerprint sensors.
- GUC 100 (Gafurov et al., 2010): is a semi-public database that can be used by researchers at Gjovik University premises. In total, it contains 72,000 fingerprint images collected over several months from 100 subjects providing fingerprints of all ten fingers on six different scanners.
- Unconstrained Fingerphoto Database (Chopra et al., 2018): contains 3,450 fingerphoto images (of 115 subjects) captured using smartphone camera in unconstrained environments.

Some multimodal datasets also exist where fingerprint is one of the modalities: the following datasets are recommended for the design and test of multimodal systems:

- *MCYT Bimodal Database* (Ortega-Garcia et al., 2003; Simon-Zorita et al., 2003): fingerprint and signature modalities from 330 subjects.
- *BIOMET Multimodal Database* (Garcia-Salicetti et al., 2003): speech, face, hand, fingerprint and signature from 91 subjects.

Table 4.3 An overview of fingerprint recognition evaluation campaigns. FVC-onGoing website is available at: <https://biolab.csr.unibo.it/fvcongoing>. For more information on NIST evaluations visit: <https://www.nist.gov/programs-projects/fingerprint-recognition>

	Evaluation Campaign	Year(s)	Task(s)	Notes
University of Bologna	FVC 2000	2000	Fingerprint verification	Four databases acquired for each campaign (publicly available after the competition) Live-scan Flat fingerprints (also from small area and sweep sensors) Included difficult cases (e.g., voluntarily exaggerated rotation and distortions)
	FVC 2002	2002		
NIST	FVC 2004	2004	Fingerprint verification (from image and from ISO templates), Indexing, Orientation extraction, Secure templates, etc.	Fingerprint images acquired in operational conditions using high-quality optical scanners
	FVC 2006	2006		
	FVC-onGoing	2009 (Ongoing)		
NIST	FpVTE 2003, FpVTE 2012	2003 2012	Fingerprint verification and identification	Operational fingerprint data from a variety of U.S. Government sources (including Flat, Slap, Rolled, Inked fingerprints) Large datasets (not public after the campaigns)
	PFT 2003, PFT II, PFT III	2003 2010 2019 (Ongoing)	Fingerprint verification (proprietary solutions)	Focus on proprietary (non-interoperable) commercial algorithms Flat and Rolled optical and ink impressions at 500 and 1000 dpi, including non-contact imagery
	MINEX 04, MINEX II, MINEX III	2004 2006 2016 (Ongoing)	Fingerprint verification (interoperable solutions)	Focus on interoperable minutiae extractors and matchers. MINEX II evaluated secure match-on-card solutions

- *BioSec Multimodal Database* (Fierrez-Aguilar et al., 2007): face, speech, fingerprint and iris from 250 subjects. FVC2006 DB1, DB2 and DB3 are subsets of this database.
- *BioSecure Multimodal Database* (Ortega-Garcia et al., 2010) is composed of three parts; DB1: speech and face from 1,000 subjects acquired over the internet under unsupervised conditions; DB2: speech, face, signature, fingerprint, hand and iris from 700 subjects acquired in a standard office environment; DB3: speech, face, signature, and fingerprint from 700 subjects acquired using mobile hand-held devices under two acquisition conditions (controlled/indoor and uncontrolled/outdoor).
- *VMU Multimodal Database*⁵: contains iris, fingerprints, palmprint, hand geometry, face, face video, and voice of 200–300 subjects.

The study by Li et al. (2014a) assesses the level of difficulty (LOD) of a fingerprint database according to the common area, distortion, and relative sample quality of genuine mated pairs. Experimental results on all FVC databases + PolyU HRF show that LOD correlates quite well with the EER achieved by a selection of fingerprint recognition algorithms.

4.7.2 Fingerprint Evaluation Campaigns

Comparative evaluations of fingerprint recognition algorithms have been reported by some authors (e.g., Peralta et al., 2015). These studies are certainly of interest and can provide useful insights. However, the difficulty of exactly reproducing and parametrizing third party algorithms can undermine the outcomes. For this reason, we believe that independent evaluation campaigns can better reflect the state-of-art and provide unbiased indications of the expected performance in real applications. Designers submitting their algorithms to an evaluation campaign have full control of their implementations and, at the same time, cannot overfit the benchmark because the test set is usually sequestered (Cappelli et al., 2006).

The most important evaluation campaigns for fingerprint recognition have been organized by University of Bologna and NIST. Table 4.3 provides an overview. Evaluation campaigns on latent fingerprints (such as ELFT-EFS by NIST) are discussed in Chap. 6.

A comparative summary of the performances obtained in the four FVC campaigns is given in Table 4.4.

FVC-onGoing is a web-based automated evaluation for fingerprint recognition algorithms started in 2009. As of January 2021, about 1,700 registered participants have submitted more than 7,500 algorithms to 9 different benchmarks. In image-based fingerprint verification the top performing algorithm reached an $EER = 0.010\%$ and $EER = 0.214\%$ on the standard and hard datasets, respectively; in minutiae-based fingerprint

⁵ <https://biic.wvu.edu/data-sets/multimodal-dataset>.

Table 4.4 Average accuracy (EER) of the three best performing algorithms over four different FVC databases. A direct comparison across the different competitions is not possible due to the use of databases of unequal difficulty

	DB1 (%)	DB2 (%)	DB3 (%)	DB4 (%)
FVC2000	2.30	1.39	4.34	3.38
FVC2002	0.20	0.17	0.63	0.16
FVC2004	1.61	2.32	1.34	0.81
FVC2006	5.88	0.05	1.59	0.39

verification, the corresponding EER values are 0.194% and 1.080%. This confirms that using additional features besides minutiae can improve accuracy.

FpVTE 2012 was conducted to assess the capabilities of fingerprint identification algorithms (accuracy, speed, template size, number of fingers, etc.) using operational datasets containing several million subjects (Watson et al., 2014). The most accurate fingerprint identification submission achieved:

- single index fingers: FNIR = 1.9% @ FPIR = 0.1% on a gallery of 100 K subjects.
- two index fingers: FNIR = 0.27% @ FPIR = 0.1% on a gallery of 1.6 M subjects.
- ten fingers: FNIR = 0.09% @ FPIR = 0.1% on a gallery of 5 M subjects.

As expected, using more fingers for recognition leads to increased accuracy.

4.7.3 Interoperability of Fingerprint Recognition Algorithms

As already discussed in Sect. 3.1, standards have been released to specify the format of fingerprint templates, paving the way for interoperability of algorithms designed by different developers (e.g., matching the template created with vendor A's feature extractor with vendor B's matcher). The minutiae-based standard ISO/IEC 19794-2 is the most important one, due to the wide reliance on minutiae-based systems.

The purpose of the NIST Minutiae Interoperability Exchange Test (MINEX) started in 2004 (Grother et al., 2006) was to quantify the impact of the minutiae template standard on the verification accuracy. As expected, the results demonstrated that:

- Standard templates lead to lower verification performance than proprietary templates even if the feature extractor and matcher are provided by the same vendor; the most accurate systems tested in MINEX04 nearly doubled their FNMR error for FMR = 1% (see Table 4.5) and increased their FMR error about 10 times for FNMR = 1%.
- In MINEX interoperability scenario vendor A extracts the first standard template T_A to be matched and vendor B extracts the second standard template T_B and provides

the matching algorithm to compare \mathbf{T}_A against \mathbf{T}_B . Looking at the results of the best performing algorithms evaluated in MINEX III, it can be noted that when $A \neq B$ the matching accuracy is about 5 times lower.

- The reduced accuracy due to interoperability can be compensated for by using two fingers for all authentication attempts. In such a case the performance gain is about one order of magnitude. However, using two fingers is not as convenient for the users and, in general, it results in a longer transaction time; hence it is not feasible for many applications.

Within the scope of the PIV (Personal Identity Verification) project (Grother et al., 2013), NIST has set the maximum error (both FMR and FNMR) to 1% that an algorithm can exhibit when interoperating with every other algorithm (in a group of existing interoperable algorithms) in order to be declared compliant.

4.8 Summary

Throughout this chapter several techniques for fingerprint matching have been surveyed, and the pros and cons of different approaches have been highlighted. However, an explicit answer has not been provided to the question: what is the best algorithm for matching fingerprints?

It is difficult to give a definitive answer to this question because the performance of a fingerprint recognition method involves a tradeoff among different indicators: accuracy (e.g., FMR and FNMR), efficiency (enrollment time, verification time), scalability to 1:N identification, template size, memory requirement, and so on. Different applications have different performance requirements. For example, an application may prefer a fingerprint matching algorithm that is slightly lower in accuracy but has a smaller template size over an algorithm that is more accurate but requires a large template size; specific constraints are also imposed by system security related issues (see Chap. 9).

However, if we focus on accuracy, from FVC-onGoing results⁶ we note that the best performing algorithms are still based on minutiae and operate through an initial local matching followed by a global consolidation. The first generation of global matching algorithms has been abandoned because of their higher computational complexity and low tolerance to distortion. Minutiae local structures reached their maturity with MCC, and other texture enriched descriptors. Effective methods have been introduced to explicitly deal with local distortion, including accurate dense registration techniques.

Deep-learning techniques have been demonstrated to be effective to get closer to a long-awaited result, that is, mapping a raw fingerprint image into a compact fixed-length representation without handcrafted feature extraction. Domain knowledge can be injected

⁶ Algorithms submitted to FVC-onGoing (including commercial ones) are accompanied by structured side-information about their functioning.

(only during training) by solving a side-task of minutiae detection, further demonstrating how much minutiae features are relevant for fingerprint matching. Competitive results have been demonstrated when matching full fingerprints (rolled or flats acquired by large area scanners). We expect that future research will allow to successfully apply these techniques also to partial fingerprint matching.

References

- Abe, N., & Shinzaki, T. (2015). Vectorized fingerprint representation using minutiae relation code. In *Proceedings of International Conference on Biometrics* (pp. 408–415). Phuket.
- Abhyankar, A., & Schuckers, S. (2010). Towards integrating level-3 features with perspiration pattern for robust fingerprint recognition. In *Proceedings of International Conference on Image Processing* (pp. 3085–3088). Hong Kong.
- Ahuja, R., Magnanti, T., & Orlin, J. (1993). *Network flows*. Prentice-Hall.
- Almansa, A., & Cohen, L. (2000). Fingerprint image matching by minimization of a thin-plate energy using a two-step iterative algorithm with auxiliary variables. In *Proceedings of Workshop on Applications of Computer Vision* (pp. 35–40).
- Alshehri, H., Hussain, M., Aboalsamh, H. A., & Al Zuair, M. A. (2018). Cross-sensor fingerprint matching method based on orientation, gradient, and Gabor-HoG descriptors with score level fusion. *IEEE Access*, 6, 28951–28968.
- Amornraksa, T., & Tachaphetpiboon, S. (2006). Fingerprint recognition using DCT features. *Electronics Letters*, 42(9), 522–523.
- Aravindan, A., & Anzar, S. M. (2017). Robust partial fingerprint recognition using wavelet SIFT descriptors. *Pattern Analysis and Applications*, 20(4), 963–979.
- Bal, A., El-Saba, A. M., & Alam, M. S. (2005). Enhanced fingerprint verification and identification using a Widrow cellular neural network. *Optical Engineering*, 44(3), 037201.
- Baldi, P., & Chauvin, Y. (1993). Neural networks for fingerprint recognition. *Neural Computation*, 5(3), 402–418.
- Ballard, D. H. (1981). Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 3(2), 110–122.
- Banner, C. B., & Stock, R. M. (1974). Finder, the FBI's approach to automatic fingerprint identification. In *Proceedings of Conference on Science of Fingerprints*.
- Banner, C. B., & Stock, R. M. (1975a, January). The FBI's approach to automatic fingerprint identification (Part I). U.S. Government Publication, *FBI Law Enforcement Bulletin*, 44(1).
- Banner, C. B., & Stock, R. M. (1975b, February). The FBI's approach to automatic fingerprint identification (Part II). U.S. Government Publication, *FBI Law Enforcement Bulletin*, 44(2).
- Bazen, A. M., Verwaaijen, G. T. B., Gerez, S. H., Veelenturf, L. P. J., & van der Zwaag, B. J. (2000). A correlation-based fingerprint verification system. In *Proceedings of Workshop on Circuits Systems and Signal Processing (ProRISC 2000)*.
- Bazen, A. M., & Gerez, S. H. (2003). Fingerprint matching by thin-plate spline modelling of elastic deformations. *Pattern Recognition*, 36(8), 1859–1867.
- Bebis, G., Deaconu, T., & Georgopoulos, M. (1999). Fingerprint identification using delaunay triangulation. In *Proceedings of IEEE International Conference on Intelligence, Information, and Systems (ICIS)* (pp. 452–459).

- Beleznaï, C., Ramoser, H., Wachmann, B., Birchbauer, J., Bischof, H., & Kropatsch, W. (2001). Memory-efficient fingerprint verification. In *Proceedings of International Conference on Image Processing*.
- Benhammadi, F., Amrouche, M. N., Hentous, H., Beghdad, K. B., & Aissani, M. (2007). Fingerprint matching from minutiae texture maps. *Pattern Recognition*, 40(1), 189–197.
- Bharkad, S. D., & Kokare, M. (2012). Rotated wavelet filters-based fingerprint recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 26(3), 1256008.
- Bhowmick, P., & Bhattacharya, B. B. (2004). Approximate fingerprint matching using kd-tree. In *17th Proceedings of International Conference on Pattern Recognition* (Vol. 1, pp. 544–547).
- BioLab. (2007). BioLab – University of Bologna. FVC 2006 Web Site. Retrieved November 27, 2008 from <http://bias.csr.unibo.it/fvc2006>.
- Bouchaffra, D., & Amira, A. (2008). Structural hidden Markov models for biometrics: Fusion of face and fingerprint. *Pattern Recognition*, 41(3), 852–867.
- Bringer, J., & Despiegel, V. (2010). Binary feature vector fingerprint representation from minutiae vicinities. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems (BTAS)* (pp. 1–6). Washington, DC.
- Cao, K., Liu, E., Pang, L., Liang, J., & Tian J. (2011). Fingerprint matching by incorporating minutiae discriminability. In *Proceedings of International Joint Conference on Biometrics (IJCB)* (pp. 1–6). Washington, DC.
- Cao, K., Yang, X., Chen, X., Tao, X., Zang, Y., Liang, J., & Tian, J. (2012a). Minutia handedness: A novel global feature for minutiae-based fingerprint matching. *Pattern Recognition Letters*, 33(10), 1411–1421.
- Cao, K., Yang, X., Chen, X., Zang, Y., Liang, J., & Tian, J. (2012b). A novel ant colony optimization algorithm for large-distorted fingerprint matching. *Pattern Recognition*, 45(1), 151–161.
- Cao, K., Yang, X., Tao, X., Zhang, Y., & Tian, J. (2009). A novel matching algorithm for distorted fingerprints based on penalized quadratic model. In *Proceedings of International Conference on Biometrics: Theory, Applications, and Systems (BTAS)* (pp. 1–5). Washington, DC.
- Cappelli, R., Ferrara, M., & Maltoni, D. (2010a). Minutia cylinder-code: A new representation and matching technique for fingerprint recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12), 2128–2141.
- Cappelli, R., Ferrara, M., Maltoni, D., & Tistarelli M. (2010b). MCC: A baseline algorithm for fingerprint verification in FVC-onGoing. In *Proceddings International Conference on Control Automation Robotics & Vision*, Singapore.
- Cappelli, R., Ferrara, M., & Maltoni, D. (2015). Large-scale fingerprint identification on GPU. *Information Sciences*, 306, 1–20.
- Cappelli, R., Ferrara, M., & Maltoni, D. (2018). Large scale fingerprint recognition accelerated in hardware. In M. Drahanský (Ed.), *Hand-based biometrics: Methods and technology*. IET.
- Cappelli, R., Maio, D., & Maltoni, D. (2001). Modelling plastic distortion in fingerprint images. In *2nd Proceedings of International Conference on Advances in Pattern Recognition* (pp. 369–376).
- Cappelli, R., Maio, D., Maltoni, D., Wayman, J. L., & Jain, A. K. (2006). Performance evaluation of fingerprint verification systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1), 3–18.
- Carvalho, C., & Yehia, H. (2004). Fingerprint alignment using line segments. In *Proceedings of International Conference on Biometric Authentication* (pp. 380–387).
- Chang, S. H., Cheng, F. H., Hsu, W. H., & Wu, G. Z. (1997). Fast algorithm for point pattern-matching: Invariant to translations, rotations and scale changes. *Pattern Recognition*, 30(2), 311–320.
- Chebira, A., Coelho, L. P., Sandryhaila, A., Lin, S., Jenkinson, W. G., MacSweeney, J., Hoffman, C., Cuadra, P., Jackson, C., Puschel, M., & Kovacevic, J. (2007). An adaptive multiresolution

- approach to fingerprint recognition. In *Proceedings of International Conference on Image Processing* (Vol. 1, pp. 457–460).
- Chen, Z., & Kuo, C. H. (1991). A topology-based matching algorithm for fingerprint authentication. In *25th Proceedings of International Carnahan Conference on Security Technology* (pp. 84–87).
- Chen, Y., & Jain, A. K. (2007). Dots and incipients: Extended features for partial fingerprint matching. In *Proceedings of Biometric Symposium*.
- Chen, H., Tian, J., & Yang, X. (2003). Fingerprint matching with registration pattern inspection. In *4th Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 327–334).
- Chen, X., Tian, J., & Yang, X. (2006a). A new algorithm for distorted fingerprints matching based on normalized fuzzy similarity measure. *IEEE Transactions on Image Processing*, 15(3), 767–776.
- Chen, X., Tian, J., Yang, X., & Zhang, Y. (2006b). An algorithm for distorted fingerprint matching based on local triangle feature set. *IEEE Transactions on Information Forensics and Security*, 1(2), 169–177.
- Cheng, J., & Tian, J. (2004). Fingerprint enhancement with dyadic scale-space. *Pattern Recognition Letters*, 25(11), 1273–1284.
- Cheng, J., Tian, J., & Chen, H. (2004). Fingerprint minutiae matching with orientation and ridge. In *Proceedings of International Conference on Biometric Authentication* (pp. 351–358).
- Chikkerur, S., Pankanti, S., Jea, A., Ratha, N., & Bolle, R. (2006). Fingerprint representation using localized texture features. In *Proceedings of International Conference on Pattern Recognition* (Vol. 4, pp. 521–524).
- Chikkerur, S., & Ratha, N. (2005). Impact of singular point detection on fingerprint matching performance. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 207–212).
- Choi, H., Choi, K., & Kim, J. (2011). Fingerprint matching incorporating ridge features with minutiae. *IEEE Transactions on Information Forensics and Security*, 6(2), 338–345.
- Chopra, S., Malhotra, A., Vatsa, M., & Singh, R. (2018). Unconstrained fingerphoto database. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (pp. 630–6308).
- Coetzee, L., & Botha, E. C. (1990, November). Fingerprint recognition with a neural-net classifier. In *1st Proceedings of South African Workshop on Pattern Recognition* (Vol. 1, pp. 33–40).
- Coetzee, L., & Botha, E. C. (1993). Fingerprint recognition in low quality images. *Pattern Recognition*, 26(10), 1441–1460.
- Crouzil, A., Massip-Pailhes, L., & Castan, S. (1996). A new correlation criterion based on gradient fields similarity. In *13th Proceedings of International Conference on Pattern Recognition* (pp. 632–636).
- Cui, Z., Feng, J., Li, S., Lu, J., & Zhou, J. (2018). 2-D phase demodulation for deformable fingerprint registration. *IEEE Transactions on Information Forensics and Security*, 13(12), 3153–3165.
- Cui, Z., Feng, J., & Zhou, J. (2019, June). Dense fingerprint registration via displacement regression network. In *Proceedings of International Conference Biometrics (ICB)* (pp. 1–8).
- Cui, Z., Feng, J., & Zhou, J. (2021). Dense registration and mosaicking of fingerprints by training an end-to-end network. *IEEE Transactions on Information Forensics and Security*, 16, 627–642.
- Dabouei, A., Kazemi, H., Iranmanesh, S. M., Dawson, J., & Nasrabadi, N. M. (2018). Fingerprint distortion rectification using deep convolutional neural networks. In *Proceedings of International Conference on Biometrics (ICB)*.
- Deng, H., & Huo, Q. (2005). Minutiae matching based fingerprint verification using delaunay triangulation and aligned-edge-guided triangle matching. In *5th Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 270–278).

- Dorai, C., Ratha, N. K., & Bolle, R. M. (2000). Detecting dynamic behavior in compressed fingerprint videos: Distortion. In *Proceedings of Conference Computer Vision and Pattern Recognition* (Vol. 2, pp. 320–326).
- Dorai, C., Ratha, N., & Bolle, R. M. (2004). Dynamic behavior in fingerprint videos. In N. Ratha & R. Bolle (Eds.), *Automatic fingerprint recognition systems* (pp. 67–86). Springer.
- Engelsma, J. J., Cao, K., & Jain, A. K. (2021). Learning a fixed-length fingerprint representation. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 43(6), 1981–1997.
- Feng, J. (2008). Combining minutiae descriptors for fingerprint matching. *Pattern Recognition*, 41(1), 342–352.
- Feng, J., & Cai, A. (2006). Fingerprint representation and matching in ridge coordinate system. In *18th Proceedings of International Conference on Pattern Recognition* (Vol. 4, pp. 485–488).
- Feng, J., Ouyang, Z., & Cai, A. (2006). Fingerprint matching using ridges. *Pattern Recognition*, 39(11), 2131–2140.
- Feng, J., & Zhou, J. (2011). A performance evaluation of fingerprint minutia descriptors. In *Proceedings of International Conference on Hand-based Biometrics (ICHB)*, Hong Kong.
- Fielding, K., Homer, J., & Makekau, C. (1991). Optical fingerprint identification by binary joint transform correlation. *Optical Engineering*, 30(12), 1958.
- Pierrez-Aguilar, J., Ortega-Garcia, J., Torre-Toledano, D., & Gonzalez-Rodriguez, J. (2007). BioSec baseline corpus: A multimodal biometric database. *Pattern Recognition*, 40(4), 1389–1392.
- Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24, 381–395.
- Fu, X., Liu, C., Bian, J., & Feng, J. (2012). Spectral correspondence method for fingerprint minutia matching. In *Proceedings of International Conference on Pattern Recognition* (pp. 1743–1746).
- Fu, X., Liu, C., Bian, J., Feng, J., Wang, H., & Mao, Z. (2013). Extended clique models: A new matching strategy for fingerprint recognition. In *Proceedings of International Conference on Biometrics (ICB)* (pp. 1–6). Madrid.
- Gafurov, D., Bours, P., Yang, B., & Busch, C. (2010). GUC100 multi-scanner fingerprint database for in-house (Semi-public) performance and interoperability evaluation. In *Proceedings of International Conference on Computational Science and Its Applications* (pp. 303–306). Fukuoka.
- Gamble, F. T., Frye, L. M., & Grieser, D. R. (1992). Real-time fingerprint verification system. *Applied Optics*, 31(5), 652–655.
- Garcia-Salicetti, S., Beumier, C., Chollet, G., Dorizzi, B., Les Jardins, J., Lunter, J., Ni, Y., & Petrovska-Delacretaz, D. (2003). BIOMET: A multimodal person authentication database including face, voice, fingerprint, hand and signature modalities. In *4th Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 845–853).
- Garris, M. D., & McCabe, R. M. (2000). *NIST Special Database 27, Fingerprint minutiae from latent and matching tenprint images*. U.S. National Institute of Standards and Technology.
- Germain, R., Califano, A., & Colville, S. (1997). Fingerprint matching using transformation parameters. *IEEE Computational Science and Engineering*, 4(4), 42–49.
- Ghafoor, M., Iqbal S., Tariq, S. A., Taj, I. A., & Jafri, N. M. (2018). Efficient fingerprint matching using GPU. *IET Image Processing*, 12(2), 274–284.
- Gonzales, R. C., & Woods, R. E. (2007). *Digital image processing* (3rd ed.). Prentice-Hall.
- Gowrishankar, T. R. (1989). Fingerprint identification on a massively parallel architecture. In *2nd Proceedings of Symposium on Frontiers of Massively Parallel Computation* (pp. 331–334).
- Grother, P., McCabe, M., Watson, C., Indovina, M., Salamon, W., Flanagan, P., Tabassi, E., Newton, E., & Wilson, C. (2006, March). *Performance and interoperability of the INCITS 378 fingerprint template*. NIST Research Report: NISTIR 7296.

- Grother, P., Salamon, W., & Chandramouli, R. (2021). Biometric specifications for personal identity verification. NIST Special Publication 800-76-2. Retrieved July, 2021, from <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-76-2.pdf>.
- Gryciewicz, T. J. (1995). Fingerprint identification with joint transform correlator using multiple reference fingerprints. *Proceedings of SPIE (Optical Pattern Recognition VI)*, 2237, 249–254.
- Gryciewicz, T. J. (1996). Fingerprint recognition using binary nonlinear joint transform correlators. *Optoelectronic Devices and Systems for Processing, Critical Review, CR65*.
- Gryciewicz, T. J. (1999). Techniques to improve binary joint transform correlator performance for fingerprint recognition. *Optical Engineering*, 38(1), 114–119.
- Gryciewicz, T. J., & Javidi, B. (1996). Experimental comparison of binary joint transform correlators used for fingerprint identification. *Optical Engineering*, 35(9), 2519–2525.
- Gu, J., Zhou, J., & Yang, C. (2006). Fingerprint recognition by combining global structure and local cues. *IEEE Transactions on Image Processing*, 15(7), 1952–1964.
- Gu, S., Feng, J., Lu, J., & Zhou, J. (2018). Efficient rectification of distorted fingerprints. *IEEE Transactions on Information Forensics and Security*, 13(1), 156–169.
- Guiașu, S. (1977). *Information theory with applications*. McGraw-Hill.
- Gutiérrez, P. D., Lastra, M., Herrera, F., & Benítez, J. M. (2014). A high performance fingerprint matching system for large databases based on GPU. *IEEE Transactions on Information Forensics and Security*, 9(1), 62–71.
- Hao, Y., Tan, T., & Wang, Y. (2002). Fingerprint matching based on error propagation. In *Proceedings of International Conference on Image Processing* (Vol. 1, pp. 273–276).
- Hao, F., Anderson, F., & Daugman, J. (2006). Combining crypto with biometrics. *IEEE Transactions on Computers*, 55(9), 1081–1088.
- Hatano, T., Adachi, T., Shigematsu, S., Morimura, H., Onishi, S., Okazaki, Y., & Kyuragi, H. (2002). A fingerprint verification algorithm using the differential matching rate. In *16th Proceedings of International Conference on Pattern Recognition* (Vol. 3, pp. 799–802).
- He, Y., Kohno, R., & Imai, H. (1993). A fast automatic fingerprint identification method based on a weighted-mean of binary image. *IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences*, E76-A(9), 1469–1482.
- He, Y., Tian, J., Luo, X., & Zhang, T. (2003a). Image enhancement and minutiae matching in fingerprint verification. *Pattern Recognition Letters*, 24(9), 1349–1360.
- He, Y., Tian, J., Ren, Q., & Yang, X. (2003b). Maximum-likelihood deformation analysis of different-sized fingerprints. In *4th Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 421–428).
- He, Y., Tian, J., Li, L., Chen, H., & Yang, X. (2006). Fingerprint matching based on global comprehensive similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(6), 850–862.
- He, X., Tian, J., Li, L., He, Y., & Yang, X. (2007). Modeling and analysis of local comprehensive minutiae relation for fingerprint matching. *IEEE Transaction on Systems, Man, and Cybernetics, Part B*, 37(5), 1204–1211.
- Hoshino, Y., Asai, K., Kato, Y., & Kiji, K. (1980). Automatic reading and matching for single-fingerprint identification. In *65th Proceedings of International Association for Identification Annual Educational Conference* (pp. 1–7).
- Hrechak, A., & McHugh, J. (1990). Automated fingerprint recognition using structural matching. *Pattern Recognition*, 23(8), 893–904.
- Hu, Z., Li, D., Isshiki, T., & Kunieda, H. (2017). Hybrid minutiae descriptor for narrow fingerprint verification. *IEICE Transactions on Information and Systems*, E100D(3), 546–555.
- Huang, K., & Aviyente, S. (2004a). Choosing best basis in wavelet packets for fingerprint matching. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 1249–1252).

- Huang, K., & Aviyente, S. (2004b). Fingerprint verification based on wavelet subbands. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification*.
- Huvanandana, S., Kim, C., & Hwang, J. N. (2000). Reliable and fast fingerprint identification for security applications. In *Proceedings of International Conference on Image Processing*.
- Isenor, D. K., & Zaky, S. G. (1986). Fingerprint identification using graph matching. *Pattern Recognition*, 19(2), 113–122.
- Ito, K., Morita, A., Aoki, T., Higuchi, T., Nakajima, H., & Kobayashi K. (2005). A fingerprint recognition algorithm using phase-based image matching for low-quality fingerprints. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 33–36).
- Ito, K., Morita, A., Aoki, T., Nakajima, H., Kobayashi, K., & Higuchi, T. (2006). A fingerprint recognition algorithm combining phase-based image matching and feature-based matching. In *Proceedings of International Conference on Biometrics*. LNCS (Vol. 3832, pp. 316–325).
- Izadi, M. H., Mirmohamadsadeghi, L., & Drygajlo, A (2012). Introduction of cylinder quality measure into minutia cylinder-code based fingerprint matching. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems (BTAS)* (pp. 353–358). Arlington, VA.
- Jain, A. K., Hong, L., & Bolle, R. (1997). On-line fingerprint verification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4), 302–313.
- Jain, A. K., Hong, L., & Pankanti, S. (2000, February). Biometrics: Promising frontiers for emerging identification market. In *Communications of the ACM* (pp. 91–98).
- Jain, A. K., Pankanti, S., Prabhakar, S., & Ross, A. (2001). Recent advances in fingerprint verification. In *3rd Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 182–191).
- Jain, M. D., Pradeep, N. S., Prakash, C., & Raman, B. (2006). Binary tree based linear time fingerprint matching. In *Proceedings of International Conference on Image Processing* (pp. 309–312).
- Jain, A. K., Chen, Y., & Demirkus, M. (2007). Pores and ridges: High-resolution fingerprint matching using Level 3 features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(1), 15–27.
- Jea, T. Y., & Govindaraju, V. (2005). A minutia-based partial fingerprint recognition system. *Pattern Recognition*, 38(10), 1672–1684.
- Jia, J., Cai, L., Lu, P., & Lu, X. (2007). Fingerprint matching based on weighting method and the SVM. *Neurocomputing*, 70(4–6), 849–858.
- Jia, X., Yang, X., Zang, Y., Zhang, N., & Tian, J. (2012). A cross-device matching fingerprint database from multi-type sensors. In *Proceedings of International Conference on Pattern Recognition (ICPR)* (pp. 3001–3004). Tsukuba.
- Jiang, X., & Yau, W. Y. (2000). Fingerprint minutiae matching based on the local and global structures. In *15th Proceedings of International Conference on Pattern Recognition* (Vol. 2, pp. 1042–1045).
- Jin, Z., Lim, M., Teoh, A. B. J., Goi, B., & Tay, Y. H. (2016). Generating fixed-length representation from minutiae using kernel methods for fingerprint authentication. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(10), 1415–1428.
- Khalil, M. S., Muhammad, D., Khan, M. K., & AL-Nuzaili Q. (2009). Fingerprint verification using fingerprint texture. In *Proceedings of Interenational Symposium on Signal Processing and Information Technology (ISSPIT)* (pp. 591–594). Ajman.
- Kho, J. B., Teoh, A. B. J., Lee, W., & Kim, J. (2020). Bit-string representation of a fingerprint image by normalized local structures. *Pattern Recognition*, 103, 107323.
- Kobayashi, T. (1992). A fingerprint image recognition method for network user identification. In *4th Proceedings of International Conference on Computing and Information* (pp. 369–372).

- Kobayashi, Y., & Toyoda, H. (1999). Development of an optical joint transform correlation system for fingerprint recognition. *Optical Engineering*, 38(7), 1205–1210.
- Kovacs-Vajna, Z. M. (2000). A fingerprint verification system based on triangular matching and dynamic time warping. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11), 1266–1276.
- Kryszczuk, K. M., Morier, P., & Drygajlo, A. (2004). Study of the distinctiveness of level 2 and level 3 features in fragmentary fingerprint comparison. In *Proceedings of ECCV Workshop on Biometric Authentication* (pp. 124–133).
- Kulkarni, J. V., Patil, B. D., & Holambe, R. S. (2006). Orientation feature for fingerprint matching. *Pattern Recognition*, 39(8), 1551–1554.
- Kumar, B. V. K. V., Savvides, M., Xie, C., Venkataramani, K., Thornton, J., & Mahalanobis, A. (2004). Biometric verification with correlation filters. *Applied Optics*, 43(2), 391–402.
- Kwon, D., Yun, I. D., Kim, D. H., & Lee, S. U. (2006). Fingerprint matching method using minutiae clustering and warping. In *18th Proceedings of International Conference on Pattern Recognition* (Vol. 4, pp. 525–528).
- Kwon, D., Yun, I. D., & Lee, S. U. (2007). A robust warping method for fingerprint matching. In *Proceedings of Conference Computer Vision and Pattern Recognition*.
- Lal, A., Zang, D. Y., & Millerd, J. (1999). Laser-diode-based joint transform correlator for fingerprint identification. *Optical Engineering*, 38(1), 69–75.
- Lan, S., Guo, Z., & You, J. (2019). A non-rigid registration method with application to distorted fingerprint matching. *Pattern Recognition*, 95, 48–57.
- Lan, S., Guo, Z., & You, J. (2020). Pre-registration of translated/distorted fingerprints based on correlation and the orientation field. *Information Sciences*, 520, 292–304.
- Lastra, M., Carabaño, J., Gutiérrez, P. D., Benítez, J. M., & Herrera, F. (2015). Fast fingerprint identification using GPUs. *Information Sciences*, 301, 195–214.
- Le, T. V., Cheung, K. Y., & Nguyen, M. H. (2001). A fingerprint recognizer using fuzzy evolutionary programming. In *Proceedings of International Conference on System Sciences*.
- Lee, D., Choi, K., & Kim, J. (2002). A robust fingerprint matching algorithm using local alignment. In *16th Proceedings of International Conference on Pattern Recognition* (Vol. 3, pp. 803–806).
- Leordeanu, M., & Hebert, M. (2005). A spectral technique for correspondence problems using pairwise constraints. In *Proceedings of International Conference on Computer Vision (ICCV)* (Vol. 2, pp. 1482–1489).
- Li, C., Fu, B., Li, J., & Yang, X. (2012). Texture-based fingerprint recognition combining directional filter banks and wavelet. *International Journal of Pattern Recognition and Artificial Intelligence*, 26(4), 1–20.
- Li, S., Kim, H., Jin, C., Elliott, S., & Ma, M. (2014a). Assessing the level of difficulty of fingerprint datasets based on relative quality measures. *Information Sciences*, 268, 122–132.
- Li, J., Tulyakov, S., & Govindaraju, V. (2014b). Improved local correlation method for fingerprint matching. In *Proceedings of International Symposium on Computing and Networking*.
- Li, Z., & Zhang, D. (1984). A fingerprint recognition system with micro-computer. In *7th Proceedings of International Conference on Pattern Recognition* (pp. 939–941).
- Liang, X., & Asano, T. (2006). Fingerprint matching using minutia polygons. In *18th Proceedings of International Conference on Pattern Recognition* (Vol. 1, pp. 1046–1049).
- Liang, X., Bishnu, A., & Asano, T. (2007). A robust fingerprint indexing scheme using minutia neighborhood structure and low-order delaunay triangles. *IEEE Transactions on Information Forensics and Security*, 2(4), 721–733.
- Lin, C., & Kumar, A. (2019). A CNN-based framework for comparison of contactless to contact-based fingerprints. *IEEE Transactions on Information Forensics and Security*, 14(3), 662–676.

- Lindoso, A., Entrena, L., Liu-Jimenez, J., & San Millan, E. (2007). Correlation-based fingerprint matching with orientation field alignment. In *Proceedings of International Conference on Biometrics*. LNCS (Vol. 4642, pp. 713–721).
- Liu, J. H., Lin, C. H., Osterburg, J. W., & Nichol, J. D. (1982). Fingerprint comparison II: On the development of a single fingerprint filing and searching system. *Journal of Forensic Sciences*, 27(2), 305–317.
- Liu, M., Jiang, X., & Kot, A. C. (2004). Fingerprint reference point detection. In *1st Proceedings of International Conference on Biometric Authentication*. LNCS (Vol. 3072, pp. 272–279).
- Liu, L., Jiang, T., Yang, J., & Zhu, C. (2006). Fingerprint registration by maximization of mutual information. *IEEE Transactions on Image Processing*, 15(5), 1100–1110.
- Liu, Y., Li, D., Isshiki, T., & Kunieda, H. (2010a). A novel similarity measurement for minutiae-based fingerprint verification. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems (BTAS)* (pp. 1–6). Washington, DC.
- Liu, F., Zhao, Q., Zhang, L., & Zhang, D. (2010b). Fingerprint pore matching based on sparse representation. In *Proceedings of International Conference on Pattern Recognition* (pp. 1630–1633). Istanbul.
- Liu, F., Zhao, Q., & Zhang, D. (2011). A novel hierarchical fingerprint matching approach. *Pattern Recognition*, 44(8), 1604–1613.
- Liu, F., Zhao, Y., Liu, G., & Shen, L. (2020). Fingerprint pore matching using deep features. *Pattern Recognition*, 102, 107208.
- Lumini, A., & Nanni, L. (2008). Advanced methods for two-class pattern recognition problem formulation for minutiae-based fingerprint verification. *Pattern Recognition Letters*, 29(2), 142–148.
- Luo, Y., Feng, J., & Zhou, J. (2014). Fingerprint matching based on global minutia cylinder code. In *Proceedings of International Joint Conference on Biometrics* (pp. 1–8).
- Luo, X., Tian, J., & Wu, Y. (2000). A minutia matching algorithm in fingerprint verification. In *15th Proceedings of International Conference on Pattern Recognition* (Vol. 4, pp. 833–836).
- Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L., & Jain, A. K. (2002a). FVC2000: Fingerprint verification competition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3), 402–412.
- Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L., & Jain, A. K. (2002b). FVC2002: Second fingerprint verification competition. In *16th Proceedings of International Conference on Pattern Recognition*.
- Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L., & Jain, A. K. (2004). FVC2004: Third fingerprint verification competition. In *1st Proceedings of International Conference on Biometric Authentication*. LNCS (Vol. 3072, pp. 1–7).
- Malathi, S., & Meena, C. (2011). Improved partial fingerprint matching based on score level fusion using pore and SIFT features. In *Proceedings of International Conference on Process Automation, Control and Computing* (pp. 1–4). Coimbatore.
- Mansukhani, P., & Govindaraju, V. (2008). Selecting optimal classification features for SVM-based elimination of incorrectly matched minutiae. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification V*.
- Mansukhani, P., Tulyakov, S., & Govindaraju, V. (2007). Using support vector machines to eliminate false minutiae matches during fingerprint verification. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification IV*.
- Mathur, S., Vjay, A., Shah, J., Das, S., & Malla, A. (2016). Methodology for partial fingerprint enrollment and authentication on mobile devices. In *Proceedings of International Conference on Biometrics (ICB)* (pp. 1–8). Halmstad.

- McMahon, D., Johnson, G. L., Teeter, S. L., & Whitney, C. G. (1975). A hybrid optical computer processing technique for fingerprint identification. *IEEE Transaction Computer*, *C-24*(4), 358–369.
- Meenen, P., Ashrafi, A., & Adhami, R. (2006). The utilization of a Taylor series-based transformation in fingerprint verification. *Pattern Recognition Letters*, *27*(14), 1606–1618.
- Melin, P., Bravo, D., & Castillo, O. (2005). Fingerprint recognition using modular neural networks and fuzzy integrals for response integration. In *Proceedings of International Joint Conference on Neural Networks* (Vol. 4, pp. 2589–2594).
- Millard, K. (1975). An approach to automatic retrieval of latent fingerprints. In *Proceedings of International Carnahan Conference on Electronic Crime Countermeasures* (pp. 45–51).
- Millard, K. (1983). Developments on automatic fingerprint recognition. In *17th Proceedings of International Carnahan Conference on Security Technology* (pp. 173–178).
- Moayer, B., & Fu, K. (1986). A tree system approach for fingerprint pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *8*(3), 376–388.
- Nachar, R., Inaty, E., Bonnin, P. J., & Alayli, Y. (2020). Hybrid minutiae and edge corners feature points for increased fingerprint recognition performance. *Pattern Analysis and Applications*, *23*(1), 213–224.
- Nandakumar, K. (2012). Fingerprint matching based on minutiae phase spectrum. In *Proceedings of International Conference on Biometrics (ICB)* (pp. 216–221). New Delhi.
- Nanni, L., & Lumini, A. (2007). A hybrid wavelet-based fingerprint matcher. *Pattern Recognition*, *40*(11), 3146–3151.
- Nanni, L., & Lumini, A. (2008) Local binary patterns for a hybrid fingerprint matcher. *Pattern Recognition*, *41*(11), 3461–3466.
- Ng, G. S., Tong, X., Tang, X., & Shi, D. (2004). Adjacent orientation vector based fingerprint minutiae matching system. In *17th Proceedings of International Conference on Pattern Recognition* (Vol. 1, pp. 528–531).
- Nilsson, K., & Bigun, J. (2001). Using linear symmetry features as a pre-processing step for fingerprint images. In *3rd Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 247–252).
- Nilsson, K., & Bigun, J. (2005). Registration of fingerprints by complex filtering and by 1D projections of orientation images. In *5th Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 171–183).
- Novikov, S., & Ushmaev, O. (2004). Registration and modeling of elastic deformations of fingerprints. In *Proceedings of Workshop on Biometric Authentication (in ECCV 2004)*. LNCS (Vol. 3087, pp. 80–88).
- Novikov, S., & Ushmaev, O. (2005). Principal deformations of fingerprints. In *5th Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 250–259).
- Ortega-Garcia, J., Fierrez-Aguilar, J., Simon, D., Gonzalez, J., Faundez-Zanuy, M., Espinosa, V., Satue, A., Hernaez, I., Igarza, J., Vivaracho, C., Escudero, D., & Moro, Q. (2003). MCYT baseline corpus: A bimodal biometric database. *IEE Proceedings on Vision, Image and Signal Processing*, *150*(6), 395–401.
- Ortega-Garcia, J., Fierrez, J., Alonso-Fernandez, F., Galbally, J., Freire, M. R., Gonzalez-Rodriguez, J., Garcia-Mateo, C., Alba-Castro, J., Gonzalez-Agulla, E., Otero-Muras, E., Garcia-Salicetti, S., Allano, L., Ly-Van, B., Dorizzi, B., Kittler, J., Bourlai, T., Poh, N., Deravi, F., Ng, M. N. R., Fairhurst, M., Hennebert, J., Humm, A., Tistarelli, M., Brodo, L., Richiardi, J., Drygajlo, A., Ganster, H., Sukno, F.M., Pavani, S., Frangi, A., Akarun, L., & Savran, A. (2010). The multisensor scenario multienvironment BioSecure multimodal database (BMDB). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(6), 1097–1111.

- Ouyang, Z., Feng, J., Su, F., & Cai, A. (2006). Fingerprint matching with rotation-descriptor texture features. In *18th Proceedings of International Conference on Pattern Recognition* (Vol. 4, pp. 417–420).
- Park, U., Pankanti, S., & Jain, A. K. (2008). Novel fingerprint verification system using SIFT. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification V*.
- Parziale, G., & Niel, A. (2004). A fingerprint matching using minutiae triangulation. In *1st Proceedings of International Conference on Biometric Authentication*. LNCS (Vol. 3072, pp. 241–248).
- Pasha Hosseini, A., Zhdanov, R., & Ushveridze, A. (2017). An unsupervised 2D point-set registration algorithm for unlabeled feature points: Application to fingerprint matching. *Pattern Recognition Letters*, 100, 137–143.
- Paulino, A. A., Feng, J., & Jain, A. K. (2013). Latent fingerprint matching using descriptor-based hough transform. *IEEE Transactions on Information Forensics and Security*, 8(1), 31–45.
- Peralta, D., Galar, M., Triguero, I., Paternain D., García, S., Barrenechea, E., Benítez, J. M., Bustince, H., & Herrera, F. (2015). A survey on fingerprint minutiae-based local matching for verification and identification: Taxonomy and experimental evaluation. *Information Sciences*, 315, 67–87.
- Peralta, D., García, S., Benítez, J. M., & Herrera, F. (2017). Minutiae-based fingerprint matching decomposition: Methodology for big data frameworks. *Information Sciences*, 408, 198–212.
- Peralta, D., Triguero, I., Sanchez-Reillo, R., Herrera, F., & Benítez, J. M. (2014). Fast fingerprint identification for large databases. *Pattern Recognition*, 47(2), 588–602.
- Petillot, Y., Guibert, L., & de Bougrenet, J. L. (1996). Fingerprint recognition using a partially rotation invariant composite filter in a FLC joint transform correlator. *Optics Communications*, 126, 213–219.
- Prabhakar, R. V. S. N., & Rao, K. (1989). A parallel algorithm for fingerprint matching. In *Proceedings of Tencon Conference* (pp. 373–376).
- Qi, J., & Wang, Y. (2005). A robust fingerprint matching method. *Pattern Recognition*, 38(10), 1665–1671.
- Quek, C., Tan, K. B., & Sagar, V. K. (2001). Pseudo-outer product based neural network fingerprint verification system. *Neural Networks*, 14, 305–323.
- Ramos, D., Krish, R. P., Fierrez, J., & Meuwly, D. (2017a). From biometric scores to forensic likelihood ratios. In M. Tistarelli & C. Champod (Eds.), *Handbook of biometrics for forensic science*. Springer.
- Ramos, D., Haraksim, R., & Meuwly, D. (2017b). Likelihood ratio data to report the validation of a forensic fingerprint evaluation method. *Data in Brief*, 10, 75–92.
- Ranade, A., & Rosenfeld, A. (1993). Point pattern matching by relaxation. *Pattern Recognition*, 12(2), 269–275.
- Ratha, N. K., & Bolle, R. M. (1998). Effect of controlled image acquisition of fingerprint matching. In *14th Proceedings of International Conference on Pattern Recognition*.
- Ratha, N. K., Chen, S. Y., & Jain, A. K. (1995). Adaptive flow orientation-based feature extraction in fingerprint images. *Pattern Recognition*, 28(11), 1657–1672.
- Ratha, N. K., Pandit, V. D., Bolle, R. M., & Vaish, V. (2000). Robust fingerprint authentication using local structural similarity. In *Proceedings of Workshop on Applications of Computer Vision* (pp. 29–34).
- Ratha, N. K., Rover, D., & Jain, A. K. (1996a). Fingerprint matching on splash 2. In D. Buell, J. Arnold & W. Kleinfelder (Eds.), *Splash 2: FPGAS in a custom computing machine* (pp. 117–140). IEEE Computer Society Press.
- Ratha, N. K., Karu, K., Chen, S., & Jain, A. K. (1996b). A real-time matching system for large fingerprint databases. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8), 799–813.

- Roberge, D., Soutar, C., & Kumar, B. V. K. V. (1999). Optimal trade-off filter for the correlation of fingerprints. *Optical Engineering*, 38(1), 108–113.
- Roddy, A., & Stosz, J. (1997). Fingerprint features: Statistical-analysis and system performance estimates. *Proceedings of the IEEE*, 85(9), 1390–1421.
- Rodolfo, J., Rajbenbach, H., & Huignard, J. (1995). Performance of a photo-refractive joint transform correlator for fingerprint identification. *Optical Engineering*, 34(4), 1166–1171.
- Rosenfeld, A., & Kak, A. (1976). *Digital picture processing*. Academic.
- Ross, A., Dass, S. C., & Jain, A. K. (2005). A deformable model for fingerprint matching. *Pattern Recognition*, 38(1), 95–103.
- Ross, A., Dass, S. C., & Jain, A. K. (2006). Fingerprint warping using ridge curve correspondences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1), 19–30.
- Ross, A., Jain, A. K., & Reisman, J. (2003). A hybrid fingerprint matcher. *Pattern Recognition*, 36(7), 1661–1673.
- Ross, A., Reisman, J., & Jain, A. K. (2002). Fingerprint matching using feature space correlation. In *Proceedings of Workshop on Biometric Authentication (in ECCV 2002)*. LNCS (Vol. 2359, pp. 48–57). Springer.
- Rowe, R. K. (2007). Biometrics based on multispectral skin texture. In *Proceedings of International Conference on Biometrics*. LNCS (Vol. 4642, pp. 1144–1153).
- Sankaran, A., Vatsa, M., & Singh, R. (2015). Multisensor optical and latent fingerprint database. *IEEE Access*, 3, 653–665.
- Senior, A. W., & Bolle, R. (2001). Improved fingerprint matching by distortion removal. *IEICE Transactions on Information and Systems (Special Issue on Biometrics)*, E84-D(7), 825–832.
- Sha, L., & Tang, X. (2004). Orientation-improved minutiae for fingerprint matching. In *Proceedings of International Conference on Pattern Recognition* (Vol. 4, pp. 432–435).
- Sha, L., Zhao, F., & Tang, X. (2003). Improved fingercode for filterbank-based fingerprint matching. In *Proceedings of International Conference on Image Processing* (Vol. 3, pp. 895–898).
- Sha, L., Zhao, F., & Tang, X. (2005). Fingerprint matching using minutiae and interpolation-based square tessellation fingercode. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 41–44).
- Sha, L., Zhao, F., & Tang, X. (2006). Minutiae-based fingerprint matching using subset combination. In *18th Proceedings of International Conference on Pattern Recognition* (Vol. 4, pp. 566–569).
- Shabrina, N., Isshiki, T., & Kunieda H. (2016). Fingerprint authentication on touch sensor using phase-only correlation method. In *Proceedings of International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES)*.
- Sheng, W., Howells, G., Fairhurst, M. C., & Deravi, F. (2007). A memetic fingerprint matching algorithm. *IEEE Transactions on Information Forensics and Security*, 2(3), 402–412.
- Sheng, W., Howells, G., Fairhurst, M. C., Deravi, F., & Harmer, K. (2009). Consensus fingerprint matching with genetically optimised approach. *Pattern Recognition*, 42(7), 1399–1407.
- Shuai, X., Zhang, C., & Hao, P. (2007). The optimal ROS-based symmetric phase-only filter for fingerprint verification. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 381–384).
- Si, X., Feng, J., Zhou, J., & Luo, Y. (2015). Detection and rectification of distorted fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3), 555–568.
- Si, X., Feng, J., Yuan, B., & Zhou, J. (2017). Dense registration of fingerprints. *Pattern Recognition*, 63, 9252–9260.
- Simon-Zorita, D., Ortega-Garcia, J., Sanchez-Asenjo, M., & Rodriguez, J. G. (2003). Minutiae-based enhanced fingerprint verification assessment relying on image quality factors. In *Proceedings of International Conference on Image Processing* (Vol. 3, pp. 891–894).

- Singh, V. K., Gyergyek, L., & Pavesic, N. (1977). Feature recognition and classification in fingerprint patterns. In *Proceedings of International Carnahan Conference on Electronic Crime Countermeasures* (pp. 241–248).
- Sjogaard, S. (1992). Discrete neural networks and fingerprint identification. In *Proceedings of Workshop on Signal Processing* (pp. 316–322).
- Soifer, V., Kotlyar, V., Khonina, S., & Skidanov, R. (1996). Fingerprint identification using directions fields. In *13th Proceedings of International Conference on Pattern Recognition*.
- Sparrow, M., & Sparrow, P. (1985a). *A topological approach to the matching of single fingerprints: Development of algorithms for use on latent fingermarks*. U.S. Government Publication/U.S. Department of Commerce, National Bureau of Standards, Gaithersburg, MD/Washington, DC.
- Sparrow, M., & Sparrow, P. (1985b). *A topological approach to the matching of single fingerprints: Development of algorithms for use on rolled impressions*. U.S. Government Publication/U.S. Department of Commerce, National Bureau of Standards, Gaithersburg, MD/Washington, DC.
- Srinivasan, H., Srihari, S. N., Beal, M. J., Phatak, P., & Fang, G. (2006). Comparison of ROC-based and likelihood methods for fingerprint verification. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification III*.
- Starink, J. P. P., & Backer, E. (1995). Finding point correspondence using simulated annealing. *Pattern Recognition*, 28(2), 231–240.
- Stockman, G., Kopstein, S., & Benett, S. (1982). Matching images to models for registration and object detection via clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(3), 229–241.
- Stoianov, A., Soutar, C., & Graham, A. (1999). High-speed fingerprint verification using an optical correlator. *Optical Engineering*, 38(1), 99–107.
- Stosz, J. D., & Alyea, L. A. (1994). Automated system for fingerprint authentication using pores and ridge structure. In *Proceedings of SPIE (Automatic Systems for the Identification and Inspection of Humans)* (Vol. 2277, pp. 210–223).
- Sujan, V. A., & Mulqueen, M. P. (2002). Fingerprint identification using space invariant transforms. *Pattern Recognition Letters*, 23(5), 609–619.
- Sutarno, M. V., & Kistijantoro, A. I. (2017). Minutia cylinder code-based fingerprint matching optimization using GPU. In *Proceedings of International Conference on Data and Software Engineering (ICoDSE)* (pp. 1–5). Palembang.
- Szeliski, R. (2011). *Computer vision: Algorithms and applications*. Springer.
- Tan, X., & Bhanu, B. (2003). A robust two step approach for fingerprint identification. *Pattern Recognition Letters*, 24(13), 2127–2134.
- Tan, X., & Bhanu, B. (2006). Fingerprint matching by genetic algorithms. *Pattern Recognition*, 39(3), 465–477.
- Tareen, S. A. K., & Saleem, Z. (2018). A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK. In *Proceedings of International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)* (pp. 1–10).
- Tico, M., & Kuosmanen, P. (2003). Fingerprint matching using an orientation-based minutia descriptor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8), 1009–1014.
- Tico, M., Kuosmanen, P., & Saarinen, J. (2001). Wavelet domain features for fingerprint recognition. *Electronics Letters*, 37(1), 21–22.
- Ton, J., & Jain, A. K. (1989). Registering landsat images by point matching. *IEEE Transaction Geoscience Remote Sensing*, 27(5), 642–651.
- Tong, X., Huang, J., Tang, X., & Shi, D. (2005). Fingerprint minutiae matching using the adjacent feature vector. *Pattern Recognition Letters*, 26(9), 1337–1345.
- Tong, X., Liu, S., Huang, J., & Tang, X. (2008). Local relative location error descriptor-based fingerprint minutiae matching. *Pattern Recognition Letters*, 29 (3), 286–294.

- Udupa, R., Garg, G., & Sharma P. (2001). Fast and accurate fingerprint verification. In *3rd Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 192–197).
- Umeyama, S. (1991). Least-square estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4), 376–380.
- Venkataramani, K., & Kumar, B. V. K. V. (2003). Fingerprint verification using correlation filters. In *4th Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 886–894).
- Venkataramani, K., & Kumar, B. V. K. V. (2004). Performance of composite correlation filters in fingerprint verification. *Optical Engineering*, 43(8), 1820–1827.
- Venkataramani, K., Keskinoz, M., & Kumar, B. V. K. V. (2005). Soft information fusion of correlation filter output planes using support vector machines for improved fingerprint verification performance. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification II*.
- Vij, A., & Namboodiri, A. (2014). Learning minutiae neighborhoods: A new binary representation for matching fingerprints. In *Proceedings of Conference on Computer Vision and Pattern Recognition Workshops* (pp. 64–69). Columbus, OH.
- Wahab, A., Chin, S. H., & Tan, E. C. (1998). Novel approach to automated fingerprint recognition. *IEE Proceedings Vision, Image and Signal Processing*, 145(3), 160–166.
- Wan, D., & Zhou, J. (2006). Fingerprint recognition using model-based density map. *IEEE Transactions on Image Processing*, 15(6), 1690–1696.
- Wang, C., Gavrilova, M., Luo, Y., & Rokne, J. (2006) An efficient algorithm for fingerprint matching. In *18th Proceedings of International Conference on Pattern Recognition*, (Vol. 1, pp. 1034–1037).
- Wang, X., Li, J., & Niu, Y. (2007). Fingerprint matching using orientationcodes and polylines. *Pattern Recognition*, 40(11), 3164–3177.
- Watson, C. I. (1993). *NIST special database 14, fingerprint database*. U.S. National Institute of Standards and Technology.
- Watson, C. I. (1998). *NIST special database 24, digital video of live-scan fingerprint data*. U.S. National Institute of Standards and Technology.
- Watson, C. I., & Casasent, D. P. (2004a). Recognition of live-scan fingerprints with elastic distortions using correlation filters. *Optical Engineering*, 43(10), 2274–2282.
- Watson, C. I., & Casasent, D. P. (2004b). Fingerprint matching using distortion-tolerant filters. In N. Ratha & R. Bolle (Eds.), *Automatic Fingerprint Recognition Systems* (pp. 249–262). Springer.
- Watson, C., Fiumara, G., Tabassi, E., Cheng, S. L., Flanagan, P., & Salamon, W. (2021). *Fingerprint vendor technology evaluation—Evaluation of fingerprint matching algorithms*. NIST-IR 8034, 2014. Retrieved July, 2021, from <https://nvlpubs.nist.gov/nistpubs/ir/2014/NIST.IR.8034.pdf>.
- Watson, C. I., Grother, P. J., & Casasent D. P. (2000). *Distortion-tolerant filter for elastic-distorted fingerprint matching*. Tech. Report: NIST IR 6489, National Institute of Standards and Technology, Gaithersburg, Maryland.
- Watson, C. I., & Wilson, C. L. (1992a). *NIST special database 4, fingerprint database*. U.S. National Institute of Standards and Technology.
- Watson, C.I., & Wilson, C. L. (1992b). *NIST special database 9, fingerprint database*. U.S. National Institute of Standards and Technology.
- Weber, D. M. (1992). A cost effective fingerprint verification algorithm for commercial applications. In *Proceedings of South African Symposium on Communication and Signal Processing*.
- Wei, H., Guo, M., & Ou, Z. (2006). Fingerprint verification based on multistage minutiae matching. In *18th Proceedings of International Conference on Pattern Recognition* (Vol. 2, pp. 1058–1061).

- Willis, A. J., & Myers, L. (2001). A cost-effective fingerprint recognition system for use with low-quality prints and damaged fingertips. *Pattern Recognition*, 34(2), 255–270.
- Wilson, C. L., Watson, C. I., & Paek, E. G. (1997). Combined optical and neural network fingerprint matching. *Proceedings of SPIE (Optical Pattern Recognition VIII)*, 3073, 373–382.
- Xie, X., Su, F., & Cai, A. (2006). Ridge-based fingerprint recognition. In *Proceedings of International Conference on Biometrics*. LNCS (Vol. 3832, pp. 273–279).
- Xu, W., Chen, X., & Feng J. (2007). A robust fingerprint matching approach: Growing and fusing of local structures. In *Proceedings of International Conference on Biometrics*. LNCS (Vol. 4642, pp. 134–143).
- Xu, Y., Lu, G., Lu, Y., Liu, F., & Zhang, D. (2019a). Fingerprint pore comparison using local features and spatial relations. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(10), 2927–2940.
- Xu, Y., Lu, G., Lu, Y., & Zhang, D. (2019b). High resolution fingerprint recognition using pore and edge descriptors. *Pattern Recognition Letters*, 125, 773–779.
- Xu, H., & Veldhuis, R. N. J. (2009a). Spectral representations of fingerprint minutiae subsets. In *Proceedings of International Congress on Image and Signal Processing* (pp. 1–5). Tianjin.
- Xu, H., & Veldhuis, R. N. J. (2009b). Spectral minutiae representations of fingerprints enhanced by quality data. In *Proceedings of International Conference on Biometrics: Theory, Applications, and Systems (BTAS)* (pp. 1–5). Washington, DC.
- Xu, H., Veldhuis, R. N. J., Bazen, A.M., Kevenaar, T. A. M., Akkermans, T. A. H. M., & Gokberk, B. (2009a). Fingerprint verification using spectral minutiae representations. *IEEE Transactions on Information Forensics and Security*, 4(3), 397–409.
- Xu, H., Veldhuis, R. N. J., Kevenaar, T. A. M., & Akkermans, T. A. H. M. (2009b). A fast minutiae-based fingerprint recognition system. *IEEE Systems Journal*, 3(4), 418–427.
- Xu, H., & Veldhuis, R. N. J. (2010a). Binary representations of fingerprint spectral minutiae features. In *Proceedings of International Conference on Pattern Recognition* (pp. 1212–1216). Istanbul.
- Xu, H., & Veldhuis, R. N. J. (2010b). Complex spectral minutiae representation for fingerprint recognition. In *Proceedings of CVPR Workshop on Biometrics* (pp. 1–8).
- Yager, N., & Amin, A. (2004). Evaluation of fingerprint orientation field registration algorithms. In *17th Proceedings of International Conference on Pattern Recognition* (Vol. 4, pp. 641–644).
- Yager, N., & Amin, A. (2006a). Dynamic registration selection for fingerprint verification. *Pattern Recognition*, 39(11), 2141–2148.
- Yager, N., & Amin, A. (2006b). Fingerprint alignment using a two stage optimization. *Pattern Recognition Letters*, 27(5), 317–324.
- Yahagi, H., Igaki, S., & Yamagishi, F. (1990). Moving-window algorithm for fast verification. In *Proceedings of Southeastcon Conference* (pp. 343–348).
- Yamazaki, M., Li, D., Isshiki, T., & Kunieda, H. (2015). SIFT-based algorithm for fingerprint authentication on smartphone. In *Proceedings of International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES)* (pp. 1–5).
- Yin, Y., Zhao, B., & Yang, X. (2005). An on-line template improvement algorithm. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification II*.
- Yu, K. D., Na, S., & Choi, T. Y. (2005). A fingerprint matching algorithm based on radial structure and a structure-rewarding scoring strategy. In *5th Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 656–664).
- Zang, Y., Yang, X., Jia, X., Zhang, N., Tian, J., & Zhao, J. (2013). Evaluation of minutia cylinder-code on fingerprint cross-matching and its improvement with scale. In *Proceedings of International Conference on Biometrics (ICB)* (pp. 1–6). Madrid.
- Zhang, G., Huang, X., Li, S., & Wang, Y. (2004). *Boosting Local Binary Pattern (LBP)-Based Face Recognition*. *Sinobiometrics 2004*. LNCS (Vol. 3338, pp. 179–186).

- Zhang, F., Xin, S., & Feng, J. (2019). Combining global and minutia deep features for partial high-resolution fingerprint matching. *Pattern Recognition Letters*, 119, 139–147.
- Zhang, L. H., Xu, W. L., & Chang, C. (2003). Genetic algorithm for affine point pattern matching. *Pattern Recognition Letters*, 24(3), 9–19.
- Zhang, Q., & Yan, H. (2007). Fingerprint orientation field interpolation based on the constrained delaunay triangulation. *International Journal of Information and Systems Sciences*, 3(3), 438–452.
- Zhang, Y., Yang, X., Su, Q., & Tian, J. (2007). Fingerprint recognition based on combined features. In *Proceedings of International Conference on Biometrics*. LNCS (Vol. 4642, pp. 281–289).
- Zhang, Q., Yin, Y., & Yang, G. (2016). Unmatched minutiae: Useful information to boost fingerprint recognition. *Neurocomputing*, 171, 1401–1413.
- Zhao, D., Su, F., & Cai, A. (2006). Fingerprint registration using minutia clusters and centroid structure. In *18th Proceedings of International Conference on Pattern Recognition* (Vol. 4, pp. 413–416).
- Zhao, Q., Zhang, D., Zhang, L., & Luo, N. (2010). High resolution partial fingerprint alignment using pore-valley descriptors. *Pattern Recognition*, 43(3), 1050–1061.
- Zheng, X., Wang, Y., & Zhao, X. (2007). A robust matching method for distorted fingerprints. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 377–380).
- Zhu, E., Yin, J., Hu, C., & Zhang, G. (2005). Quality estimation of fingerprint image based on neural network. In *Proceedings of International Conference on Natural Computation 2005*. LNCS (Vol. 3611, pp. 65–70).



Fingerprint Classification and Indexing

5

Abstract

This chapter focuses on fingerprint type classification and indexing techniques designed to speed up fingerprint identification on large databases. Exclusive classification techniques are first reviewed and their limitations are pointed out. The focus is then shifted to more commonly used indexing methods or multi-stage matching. Details are provided on minutiae-based and deep learning-based indexing approaches. A systematic comparison of existing techniques on well-known benchmarks is finally presented.

Keywords

Classification • Indexing • Retrieval • Hashing • Exclusive classification • Continuous classification • Multi-stage matching

5.1 Introduction

The identification of a person requires a comparison of her fingerprint with all the fingerprints in a database. This database may be very large (e.g., tens of millions of fingerprints) in many forensic and civilian applications. In such cases, the identification could exhibit an unacceptably long response time. In principle, two approaches can be adopted and possibly combined: (1) search space reduction and (2) using parallel architectures for matching. This chapter is dedicated to a review of the main techniques that fall into the first case.

Sometimes, information about sex, race, age, and other data related to the individual are available and the portion of the database to be searched can be significantly reduced. However, this information is not always accessible (e.g., criminal identification based on

latent fingerprints) and, in the general case, information intrinsic to the biometric samples has to be used for an efficient retrieval.

To achieve a significant speed-up in database search, several pre-selection techniques have been proposed and adopted that can be categorized as (1) exclusive classification and (2) indexing (also referred to as continuous classification).

Exclusive fingerprint classification is aimed at grouping a set of fingerprints into disjoint classes in a consistent and reliable way, such that different impressions of the same finger fall into the same class. Several pattern recognition studies have been devoted to this topic over the last five decades. In fact, although a coarse classification does not identify a fingerprint uniquely, it can be helpful in determining when two fingerprints do not match. In some ways, for identification systems, it was considered as a coarse-level pre-matching procedure able to speed up the database search, limiting for a given query fingerprint the number of comparisons only to the fingerprints belonging to the same class.

However, exclusive classification techniques, as methods of pre-selection in identification systems, today have only a historical value, while retaining their validity in other contexts, i.e., for anthropological studies. In fact, these techniques have some substantial limitations and do not often guarantee adequate performance. The main drawback is that the number of classes is usually small and fingerprints are unevenly distributed among them: when a single fingerprint has to be searched in a large database, exclusive classification is often not able to sufficiently narrow down the search. Furthermore, ambiguous fingerprints are a related issue. In fact, in some cases, a fingerprint can share properties with more than one class.

This led some researchers to investigate retrieval systems not based on exclusive classes. The key idea in indexing, regardless of the numerous variants and implementations, is to select an appropriate representation of the fingerprints suitable for making the search operation efficient by reducing the number of candidates. Indexing is coupled with a retrieval strategy aimed to create a list of candidates to be submitted to the matching stage. In the last three decades, several fingerprint indexing techniques have been proposed. In contrast to exclusive classification, fingerprint indexing is today an active research topic and deserves attention both for non-latent and latent cases.

Although fingerprint matching is usually performed according to local features (e.g., minutiae), fingerprint classification is generally based on global features, such as ridge structure and singularities. Fingerprint indexing schemes differ based on the local and/or global features that are used to build the index.

Section 5.2 first introduces an overview of the history of fingerprint classification and then presents a short survey of the main approaches in the literature; Sect. 5.3 discusses the metrics and dataset for exclusive classification approaches. In Sect. 5.4, we address fingerprint indexing by pointing out the difference between global features and local features-based approaches and by focusing on the most relevant techniques introduced so far; finally, in Sect. 5.5, we look at their performance.

5.2 Classification

The first fingerprint classification rules were proposed in 1823 by Purkinje (Moenssens, 1971), who classified fingerprints into nine categories (transverse curve, central longitudinal stria, oblique stripe, oblique loop, almond whorl, spiral whorl, ellipse, circle, and double whorl) according to the global ridge configurations. The first in-depth scientific study on fingerprint classification was made by Francis Galton, who divided the fingerprints into three major classes (arch, loop, and whorl) and further divided each category into subcategories (Galton, 1892). Around the same time, Juan Vucetich, an Argentine police official, developed a different system of classification; the Vucetich classification system is still used in many Spanish-speaking countries. Vucetich was also the first to make a fingerprint identification of a suspect in 1892. Ten years later, Edward Henry refined Galton's classification by increasing the number of classes (Henry, 1900); the Galton-Henry classification scheme was adopted in several countries: in fact, most of the classification schemes currently used by law enforcement agencies worldwide are variants of the Galton-Henry classification scheme. Figure 5.1 shows the five most common classes of the Galton-Henry classification scheme (arch, tented arch, left loop, right loop, and whorl):

- An arch fingerprint has ridges that enter from one side, rise to a small bump, and go out the opposite side from which they entered. Arches do not have loops or deltas.

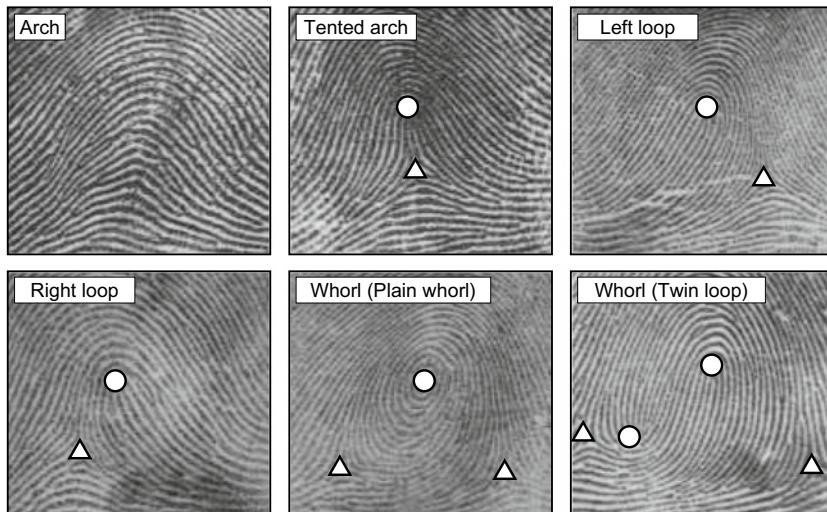


Fig. 5.1 The five commonly used fingerprint classes: two whorl fingerprints are shown (a plain whorl and a twin loop, respectively)

- A tented arch fingerprint is similar to the (plain) arch, except that at least one ridge exhibits a high curvature and one loop and one delta are present.
- A loop fingerprint has one or more ridges that enter from one side, curve back, and go out the same side they entered. Loop and delta singularities are present; the delta is assumed to be south of the loop. Loops can be further subdivided: loops that have ridges that enter and leave from the left side are called left loops and loops that have ridges that enter and leave from the right side are called right loops.
- A whorl fingerprint contains at least one ridge that makes a complete 360° path around the center of the fingerprint. Two loops (or a whorl) and two deltas can be found in whorl fingerprints. The whorl class is quite complex, and in some classification schemes, it is further divided into two categories: twin loop (or double loop) and plain whorl (see Fig. 5.1).

Fingerprint classification is a difficult pattern recognition problem due to the small inter-class variability and the large intra-class variability in the fingerprint patterns (Fig. 5.2). Moreover, fingerprint images often contain noise, which makes the classification task even more difficult (Fig. 5.3).

The selectivity of classification-based techniques strongly depends on the number of classes and the natural distribution of fingerprints in these classes. Unfortunately, the number of classes used is often small and the fingerprints are non-uniformly distributed in these classes. For example, most automatic systems use five classes (i.e., arch, tented arch, left loop, right loop, and whorl), and the natural proportion of fingerprints in these classes

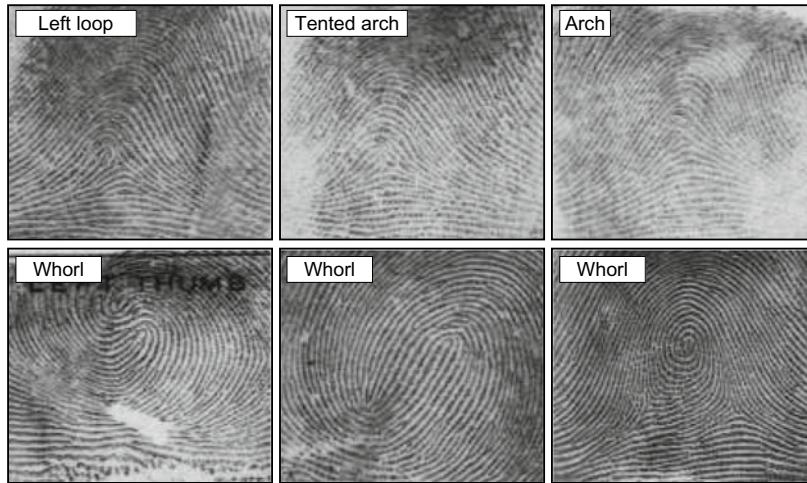


Fig. 5.2 Top row: three fingerprints belonging to different classes that have a similar appearance (small inter-class variability). Bottom row: three fingerprints belonging to the same class that have very different characteristics (large intra-class variability)



Fig. 5.3 Examples of noisy fingerprint images

is 3.7%, 2.9%, 33.8%, 31.7%, and 27.9%, respectively (Wilson et al., 1994). Furthermore, there are many “ambiguous” fingerprints, whose exclusive membership cannot be reliably stated even by human experts. Nevertheless, exclusive classification allows the efficiency of the 10-print identification¹ to be improved, because knowledge of the classes of the 10 fingerprints can be used as a code for reducing the number of comparisons at the minutiae level. On the other hand, a fingerprint classification approach does not offer sufficient selectivity for latent fingerprint searching.²

In some systems, arch and tended arch are both classified as arch because of their low occurrence. Thus, some works consider four classes. Other works divide fingerprint patterns into six categories: arc, whorl, double whorl, ulnar loop, radial loop, and peacock eye. Actually, as the number of classes increases, the classification difficulty grows as well.

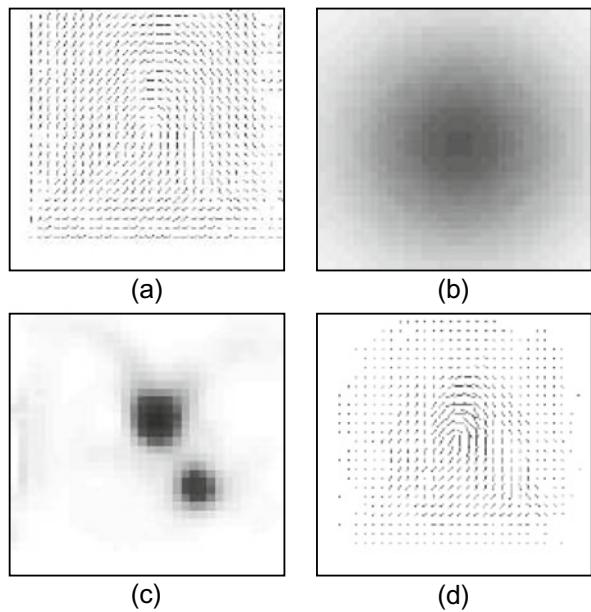
Fingerprint classification is a challenging pattern recognition problem and has attracted a significant amount of interest in the scientific community due to its importance and intrinsic difficulty, and a large number of papers have been published on this topic during the last 45 years. Although a wide variety of classification algorithms has been developed for this problem, a relatively small number of features extracted from fingerprint images have been used by most of the authors. In particular, almost all the methods are based on one or more of the following features: *ridge-line flow*, *orientation image*, *singular points*, and *Gabor filter responses*.

The ridge-line flow is usually represented as a set of curves running parallel to the ridge lines; these curves do not necessarily coincide with the fingerprint ridges and valleys, but they exhibit the same local orientation. The ridge-line flow can be traced by drawing curves locally oriented according to the orientation image (Candela et al., 1995). Most of the existing fingerprint classification approaches make use of the orientation image. This is not surprising inasmuch as such a feature, if computed with sufficient accuracy and

¹ Fingerprints from the ten fingers of an individual are compared with the corresponding fingerprints from known individuals (e.g., convicted criminals).

² A latent fingerprint (typically lifted from a crime scene) is compared to all the fingerprints in a database.

Fig. 5.4 Enhancement of the orientation image as described in Cappelli et al. (1999a): **a** orientation image, **b** Gaussian map obtained by applying the function att ; **c** irregularity map $str_{\mathbf{d}'}$, and **d** enhanced orientation image



detail, contains all the information required for the classification. Usually, the orientation image is registered with respect to the core point (see Sect. 3.5) before being processed further.

Furthermore, some authors (Cappelli et al., 1999a; Candela et al., 1995) proposed specific enhancement techniques for the orientation image \mathbf{D} which allow higher accuracy to be achieved in the successive classification stages; these techniques work by strengthening the orientation elements located in the most distinctive regions of the fingerprint image.

In Cappelli et al. (1999a), the enhancement is performed in two steps (Fig. 5.4): the effects of noise (which affects the border elements significantly) are reduced through the application of a Gaussian-like attenuation function (att), which progressively reduces the magnitude of the elements \mathbf{d} of \mathbf{D} moving from the center toward the borders, thus obtaining the new elements \mathbf{d}' :

$$\mathbf{d}' = \mathbf{d} \cdot att(\mathbf{d}, \sigma_1), \quad \text{where } att(\mathbf{d}, \sigma) = \frac{1}{\sqrt{2\pi} \cdot \sigma} e^{-(distc(\mathbf{d})^2/2\sigma^2)}, \quad (5.1)$$

$distc(\mathbf{v})$ returns the distance of \mathbf{d} from the center of the image and σ is the scale of the Gaussian function. Then, to increase the significance of the distinctive elements, the elements located in the irregular regions are strengthened by calculating, for each \mathbf{d}' :

$$str_{\mathbf{d}'} = 1 - \left| \sum_{\mathbf{d}' \in W_{5 \times 5}} \mathbf{d}' \right| \Bigg/ \sum_{\mathbf{d}' \in W_{5 \times 5}} |\mathbf{d}'| \quad (5.2)$$

$str_{\mathbf{d}'}$ is a measure of irregularity of the 5×5 neighborhood of \mathbf{d}' (denoted by $W_{5 \times 5}$). The final enhanced orientation image is made up of vectors \mathbf{d}'' such that

$$\mathbf{d}'' = \mathbf{d}' \cdot (1 + R_m \cdot \overline{str_{\mathbf{d}'}} \cdot att(\mathbf{d}', \sigma_2)), \quad (5.3)$$

where R_m is a weighting factor and $\overline{str_{\mathbf{d}'}}$ is the local average of $str_{\mathbf{d}'}$ over a 3×3 window.

Most of the existing fingerprint classification methods can be coarsely assigned to one of these categories: *rule-based*, *syntactic*, *structural*, *statistical*, *neural network-based*, and *multi-classifier* approaches. Hereafter, we provide a brief summary of the main approaches in the above categories. For a more comprehensive review, the interested reader should refer to this chapter of the previous version of this Handbook. An extended and more recent review of the literature on fingerprint classification, both from feature extraction and classification points of view, is reported in Galar et al. (2015a), where a different taxonomy is proposed. Furthermore, Galar et al. (2015b) include a deep experimental analysis of various approaches and show that combining different feature extraction models can increase the accuracy obtained by individual models.

5.2.1 Rule-Based Approaches

A fingerprint can be simply classified according to the number and the position of the singularities (see Table 5.1 and Fig. 5.1); this is the approach commonly used by human experts for manual classification; therefore, several authors proposed to adopt the same technique for automatic classification. These techniques do not require a training set to build a classification model: a fixed set of rules is defined to decide the class of the fingerprint.

Probably the progenitor scientific work of this category is Kawagoe and Tojo (1984), where the Poincaré index (see Sect. 3.5.1) is exploited to find the type and position of the singular points and a coarse classification (according to Table 5.1) is derived. Then, a finer classification is obtained by tracing the ridge-line flow. Discrimination among tented arch, left loop, and right loop is performed according to the inclination of the central trace. The authors also try to distinguish between plain whorl and twin loop; for this purpose,

Table 5.1 Singular points in the five fingerprint classes

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

two parameters (*twinness* and *flatness*) are calculated and an empirical rule is adopted to make the final decision.

Although singularity-based methods are attractive for their simplicity, some problems arise in the presence of noisy or partial fingerprints, where singularity detection can be extremely difficult. Several works addressed this aspect, e.g., Karu and Jain (1996), Ratha et al. (1996), Ballan et al. (1997), Bartesaghi et al. (2001), Wang and Xie (2004), and Klimanee and Nguyen (2004). More robust techniques are proposed in Hong and Jain (1999), Zhang and Yan (2004), and Wang and Dai (2007).

A further problem with the singularity-based approaches is that, although they may work reasonably well on *rolled* (nail-to-nail) fingerprint impressions scanned from cards, they are not suitable to be used on *dab* (live-scan) fingerprint images, because delta points are often missing in these types of images. Chong et al. (1997), Cho et al. (2000), and Jain and Minut (2002) attempted to propose solutions to the above-cited problem.

5.2.2 Syntactic Approaches

A syntactic method describes patterns by means of terminal symbols and production rules; a grammar is defined for each class and a parsing process is responsible for classifying each new pattern (Fu & Booth, 1986a, b).

Syntactic approaches were proposed by Tou and Hankley (1968) and Grasselli (1969), whose methods were based on context-free grammars, and by Verma and Chatterjee (1989), who adopted regular grammars. Other significant approaches were proposed by Moayer and Fu (1973, 1975, 1976, 1986). A related approach based on the analysis of ridge-line flow was introduced by Rao and Balck (1980). A further method combining structural and syntactic approaches was proposed by Chang and Fan (2002).

In general, due to the great diversity of fingerprint patterns, syntactic approaches require very complex grammars whose inference needs for complicated and unstable approaches; for this reason, the use of syntactic methods for fingerprint classification has been abandoned.

5.2.3 Structural Approaches

Structural approaches are based on the relational organization of low-level features into higher level structures. This relational organization is represented by means of symbolic data structures, such as trees and graphs, which allow a hierarchical organization of the information (Bunke, 1993).

The orientation image is well suited for structural representation: in fact, it can be partitioned into connected regions that are characterized by “homogeneous” orientations; these regions and the relations among them contain information useful for classification.

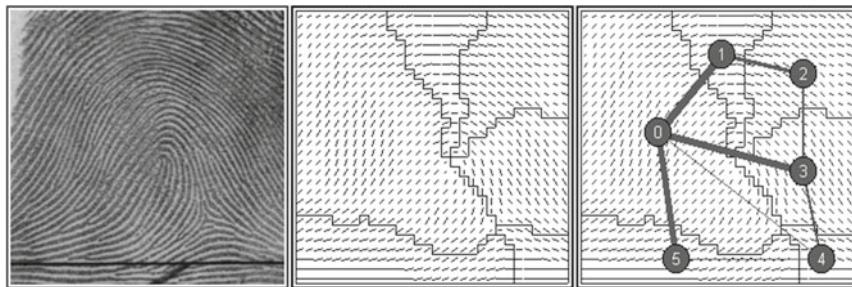


Fig. 5.5 Classification approach of Maio and Maltoni (1996): from left to right: a fingerprint image, the partitioning of its orientation image, and the corresponding relational graph. © IEEE. Reprinted, with permission, from Cappelli et al. (1999a)

This is the basic idea of the method proposed by Maio and Maltoni (1996): the orientation image is partitioned into regions by minimizing a cost function that takes into account the variance of the element orientations within each region (Fig. 5.5). An inexact graph matching technique is then used to compare the relational graphs with class-prototype graphs (see also Lumini et al., 1999).

A similar structural representation is also adopted by Yao et al. (2003) to train a recursive neural network that is combined with a Support Vector Machine classifier. Another approach based on relational graphs, but created starting from different features, is proposed by Neuhaus and Bunke (2005).

Although relational graph approaches have interesting properties (such as invariance to rotation and displacement, and the possibility of handling partial fingerprints), it is not easy to robustly partition the orientation image into homogeneous regions, especially in poor-quality fingerprints. In Cappelli et al. (1999a), a template-based matching is performed to guide the partitioning of the orientation images (Fig. 5.6): the main advantage of the approach is that, because it relies only on global structural information, it is able to deal with partial fingerprints, where sometimes, singular points are not available, and it can also work on very noisy images.

Senior (1997) adopted a Hidden Markov Model (HMM) classifier for fingerprint classification.

5.2.4 Statistical Approaches

In statistical approaches, a fixed-size numerical feature vector is derived from each fingerprint and a general-purpose statistical classifier is used for the classification. Some of the most widely adopted statistical classifiers (Jain et al., 2000) are Bayes' decision rule, k -nearest neighbor, and Support Vector Machines (SVM). Examples of their application can be found in several fingerprint classification approaches, for instance:

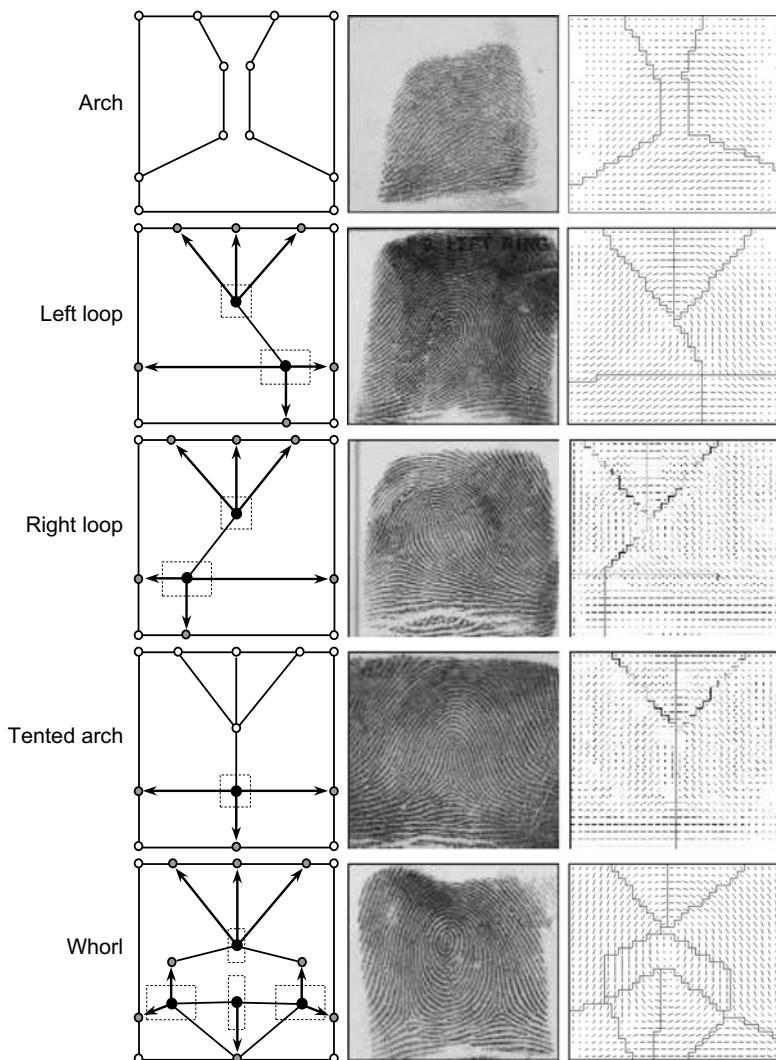


Fig. 5.6 Classification scheme of Cappelli et al. (1999a): the templates corresponding to the five classes and an example of application of each template to the orientation image of a fingerprint belonging to the corresponding class. © IEEE. Reprinted, with permission, from Cappelli et al. (1999a)

- The Bayes decision rule is adopted by Tan et al. (2005) to classify features learned by a genetic algorithm starting from the orientation image and by Hong et al. (2008) to dynamically organize a set of SVM classifiers.
- The k -nearest neighbor is exploited in Fitz and Green (1996), where wedge-ring features obtained from the hexagonal Fourier transform are used as input, and in Jain

et al. (1999), where the first step of a two-stage classification technique is performed by means of the k -nearest neighbor rule (see Sect. 5.2.6).

- SVM classifiers are used by Li et al. (2008) to classify feature vectors obtained from the coefficients of a constrained non-linear phase orientation model (see Sect. 3.3.5); SVM classifiers are often combined with other classifiers to improve the performance, such as in Yao et al. (2003), Min et al. (2006), and Hong et al. (2008); see Sect. 5.2.6.

Many approaches directly use the orientation image as a feature vector, by simply nesting its rows (see, for instance, Cappelli et al., 1999b; Candela et al., 1995). Training a classifier with such high-dimensional vectors would require large amounts of training data, memory, and computation time. For this reason, statistical dimensionality reduction techniques are often applied to reduce the dimensionality of the feature vector. The Karhunen-Loëve (KL) transform (Jolliffe, 1986) is usually adopted for this purpose, as it guarantees a good preservation of Euclidean distances between vectors (see, for instance, Wilson et al., 1994; Halici & Ongun, 1996). Another possibility is to apply an approach based on Discriminant Analysis (Jain et al., 2000) to extract a more discriminant reduced feature vector; for instance, Park and Park (2005) apply Non-linear Discriminant Analysis to the orientation image and classify the feature vectors in the reduced space through a simple nearest-centroid rule.

The KL transform, besides being used for dimensionality reduction, can also be adopted for the classification itself. In Cappelli et al. (1999b), Cappelli et al. (2000a), and Cappelli and Maio (2004), a generalization of the KL transform called MKL (which was given in a more general context in Cappelli et al., 2001) was used for representing and classifying feature vectors derived from orientation images. The underlying idea of the approach was to find, for each class, one or more KL subspaces that were well suited to represent the fingerprints belonging to that class. In Cappelli et al. (1999b), the classification was simply performed according to the minimum distance from the MKL subspaces, whereas in Cappelli and Maio (2004), the results of both minimum distance and k -nearest neighbor classifiers were reported.

5.2.5 Neural Network-Based Approaches

Initial neural network (NN) approaches are based on multilayer perceptrons and use the elements of the orientation image as input features (Hughes & Green, 1991; Bowen, 1992; Kamijo et al., 1992; Kamijo, 1993; Pal & Mitra, 1996; Shah & Sastry, 2004). Kamijo (1993) presents a pyramidal architecture constituted of several multilayer perceptrons, each of which is trained to recognize fingerprints belonging to a different class. In Bowen (1992), the location of the singularities is used together with a 20×20 orientation image for the training of two disjoint neural networks, whose outputs are passed to a third one, which produces the final classification. Jain et al. (1999) train 10 feedforward neural

networks to distinguish between each possible pair of classes (this method is described in more detail in Sect. 5.2.6).

One of the best-known neural network approaches to fingerprint classification was proposed by NIST researchers (Wilson et al., 1994), where a multilayer perceptron is used for classification after reducing the dimensionality of the feature vector as explained in the previous section. Improved versions of this method were presented in Omidvar et al. (1995) and in Candela et al. (1995), which is described in Sect. 5.2.6.

Some researchers proposed the use of self-organizing neural networks: Moscinska and Tyma (1993), Halici and Ongun (1996), and Bernard et al. (2001). A comparison of the performance of different neural network architectures for fingerprint classification (including multilayer perceptrons and Kohonen maps) used in the past is reported in Kristensen et al. (2007).

More recent NN approaches are based on convolutional NN (CNN). However, as observed by Peralta et al. (2018), there is still no comprehensive analysis of the possibilities offered by deep learning when applied to the fingerprint classification problem. Some experiments have been conducted in Peralta et al. (2018) showing that deep learning neural networks outperform other approaches. Moreover, even when CNNs are required to operate without any rejection, they still obtain better accuracy than other techniques operating with a certain rejection rate. A further application of DNN to classify low-quality fingerprint images is reported in Tertychnyi et al. (2018), where an average classification accuracy of 89.4% is reported. In Jian et al. (2020), a lightweight CNN structure based on singularity ROI (region of interest) is proposed.

5.2.6 Multiple Classifier-Based Approaches

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

Candela et al. (1995) introduced PCASYS (Pattern-level Classification Automation SYStem): a complete fingerprint classification system based on the evolution of the methods proposed in Wilson et al. (1994). Figure 5.7 shows a functional schema of PCASYS: a probabilistic neural network is coupled with an auxiliary ridge tracing module, which determines the ridge flow in the bottom part of the fingerprint; this module is specifically designed to detect whorl fingerprints. PCASYS was a milestone for successive fingerprint classification studies thanks to the availability of open-source code and because it was one of the first studies that reported precise and reproducible results on publicly available databases. Significant examples of PCASYS extensions are Yuan et al. (1998), Pattichis et al. (2001), and Senior (2001).

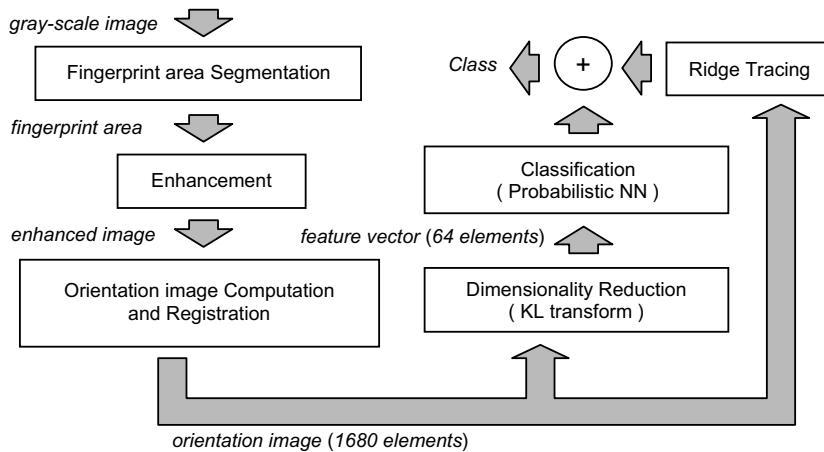


Fig. 5.7 A functional scheme of PCASYS (Candela et al., 1995). © IEEE. Reprinted, with permission, from Cappelli et al. (1999a)

Several choices are possible for the selection of the component classifiers (e.g., different classifiers trained on the same data, the same classifier trained on different data, and different input features) and for the combination strategy (from simple heuristic criteria of majority vote rule to more complex techniques that involve training an additional classifier for the final decision). Table 5.2 lists some fingerprint classification approaches that adopt different combination techniques.

Jain et al. (1999) adopt a two-stage classification strategy: a k -nearest neighbor classifier is used to find the two most likely classes from a FingerCode feature vector (see Sect. 4.6.1); then a specific neural network, trained to distinguish between the two classes, is exploited to obtain the final decision. A total of 10 neural networks is trained to distinguish between each possible pair of classes. A similar strategy is proposed by Cappelli et al. (2003), starting from different features (the orientation image) and using an MKL-based classifier to find the two most likely classes and one among 10 subspace-based classifiers for the final decision; very good results were reported by Cappelli et al. (2003) on a publicly available database (see Sect. 5.3). Shah and Sastry (2004) propose another two-stage classification strategy: at the first stage, a classifier separates arch and tented arch classes from loop and whorl; at the second stage, a classifier is trained to discriminate between arch and tented arch and three classifiers to deal with left loop, right loop, and whorl classes in a one-versus-all fashion; the authors report results obtained using this strategy with various types of classifiers (SVM, nearest neighbor, and neural network). Liu (2010) presents a fingerprint classification algorithm that uses the Adaboost method (with decision trees) to model multiple types of singularity features. Complex filters are used to detect the singularities at multiple scales and a feature vector is constructed for each scale. Fingerprint class is determined by the ensemble of the classification results at multiple

Table 5.2 Multiple classifier-based approaches

	Distinct features	Distinct classifiers	Distinct training sets	Combination strategy
Candela et al. (1995)	Yes	Yes	No	Rule-based
Jain et al. (1999)	No	Yes	No	Sequential (two stages)
Cappelli et al. (2000a)	No	Yes	Yes	Majority vote rule
Senior (2001)	Yes	Yes	No	Neural network
Marcialis et al. (2001)	Yes	Yes	No	k-nearest neighbor
Yao et al. (2003)	Yes	Yes	No	k-nearest neighbor
Cappelli et al. (2003)	No	Yes	No	Sequential (two stages)
Shah and Sastry (2004)	Yes	Yes	No	Sequential (two stages)
Hong et al. (2008)	Yes	Yes	No	Bayes' rule
Liu (2010)	Yes	Yes	No	Adaboost
Cao et al. (2013)	Yes	Yes	No	Sequential (five stages)

scales. Cao et al. (2013) propose a regularized orientation diffusion model for fingerprint orientation extraction and hierarchical classifier; the classification algorithm is composed of five cascading stages. The first stage distinguishes a majority of Arch by using complex filter responses. The second stage distinguishes a majority of Whorl by using core points and ridge-line flow classifier. In the third stage, a K-NN classifier is used to find the top two categories by using orientation field and complex filter responses. In the fourth stage, the ridge-line flow classifier distinguishes Loop from other classes except Whorl. The final classification is performed by an SVM. The classification method evaluated on the NIST DB 4 achieved a classification accuracy of 95.9% for five-class classification and 97.2% for four-class classification without any rejection.

5.2.7 Fingerprint Sub-Classification

The goal of sub-classification is to further divide some of the classes into more specific categories. This approach has been typically adopted by human experts to perform manual fingerprint searching in forensic applications. For instance, the FBI defined (Federal

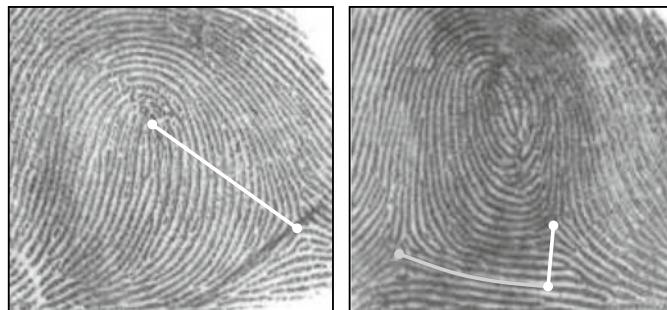


Fig. 5.8 Left: ridge counting for loop sub-classification (the number of ridges between loop and delta is 16). Right: ridge tracing and counting for whorl sub-classification (the closest point is below the rightmost delta and the number of ridges is 5)

Bureau of Investigation, 1984) a manual sub-classification procedure for loop and whorl fingerprints based on ridge counting (see Sect. 3.9); for right and left loop fingerprints, the number of ridges between the loop and delta singularities is determined: two subclasses are defined according to the number of ridges. As to whorl fingerprints, the ridge just below the leftmost delta is traced until the position closest to the rightmost delta is reached; then the number of ridges between that point and the rightmost delta is counted (Fig. 5.8).

Three sub-classes are defined depending on the number of ridges and whether the traced ridge passes over the rightmost delta. Actually, the rules are quite complicated because the sub-classification criteria also vary according to the finger (thumb, index, middle, ...).

Implementing a reliable automated fingerprint sub-classification is much more difficult than realizing a first-level classification into five classes. Therefore, it is not surprising that only a very limited number of algorithms have been proposed in the literature to address this problem (one of the very few examples is Drets and Liljenstrom (1998)). A more recent work Yong et al. (2013) proposes a new approach: first, a rotation-invariant feature is generated; then, fuzzy methods are adopted in both clustering and classification steps.

5.3 Benchmarking Fingerprint Classification Techniques

5.3.1 Metrics

The performance of a fingerprint classification system is usually measured in terms of *error rate* or *accuracy*. The error rate is computed as the ratio between the number of misclassified fingerprints and the total number of samples in the test set; the accuracy is

simply the percentage of correctly classified fingerprints:

$$\text{error rate} = \frac{\text{number of misclassified fingerprints} \times 100}{\text{total number of fingerprints}} \%. \quad (5.4)$$

$$\text{accuracy} = 100\% - \text{error rate} \quad (5.5)$$

The error rate of a classification system is generally reported as a function of the percentage of the database that the system has to search; this percentage is called *penetration rate* and can be simply computed as

$$\text{penetration rate} = \frac{\text{number of accessed fingerprints} \times 100}{\text{total number of fingerprints in the database}} \%. \quad (5.6)$$

A more detailed analysis of the behavior of a classifier can be obtained by examining the *confusion matrix*. This matrix has a row for each true class and a column for each hypothesized class; each cell at row r and column c reports how many fingerprints belonging to class r are assigned to class c . Table 5.3 shows two examples of the confusion matrix.

Fingerprint images of poor quality are often difficult to classify, even for a human expert: in many applications, it is desirable that a fingerprint classification algorithm rejects such images because this would be less damaging than a wrong decision. For this reason, several classification approaches include a rejection mechanism, which improves

Table 5.3 Confusion matrices of the results on DB4 for the approach proposed in Cappelli et al. (2003): five-class problem (on the top) and four-class problem (on the bottom)

True class	Hypothesized class				
	A	L	R	W	T
A	420	6	3	1	11
L	3	376	3	9	11
R	5	1	392	6	16
W	2	5	14	377	1
T	33	18	9	0	278

True class	Hypothesized class			
	A + T	L	R	W
A + T	782	10	17	6
L	6	373	2	4
R	7	1	381	9
W	0	4	7	391

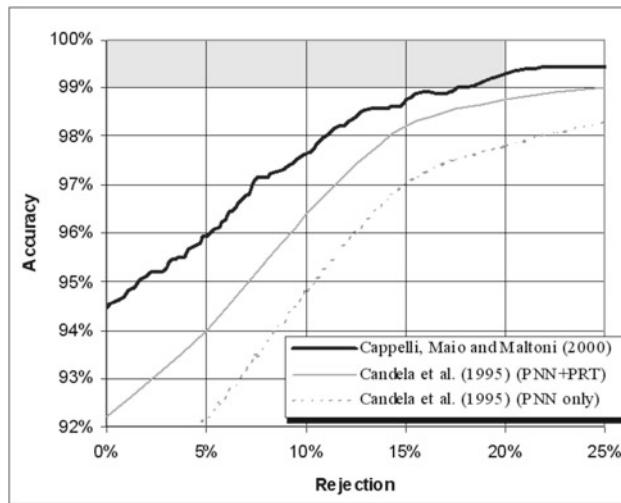


Fig. 5.9 Accuracy versus rejection curves. PCASYS performance was manually sampled from the graph reported by Candela et al. (1995). The gray area denotes the target accuracy of automatic classification set by the FBI. Probabilistic Neural Network alone (PNN only) and combined with the auxiliary Pseudo Ridge Tracing module (PNN + PRT)

the accuracy at the cost of discarding some fingerprints (i.e., classifying them as “unknown”). A confidence value is usually assigned to the classifier decision or to the fingerprint itself: the rejection simply consists of discarding fingerprints whose confidence is lower than a fixed threshold. By taking into account the rejection rate, the performance of a fingerprint classifier can be described by a graph with the rejection rate on one axis and the error rate (or the accuracy) on the other (see an example in Fig. 5.9).

5.3.2 Datasets

The early fingerprint classification systems proposed in the literature (years 1970–1990) were tested on small databases, usually collected by the authors themselves. Although the results reported on these internal databases provided an initial glimpse regarding the difficulty of the classification problem, a comparison among the various techniques was impossible and the results were not useful for tracking advances in the field. For example, in Moayer and Fu (1975) and Bowen (1992), the test sets used were two internally collected databases of 92 and 47 fingerprints, respectively: it is very difficult to deduce any conclusions from results reported on such small datasets.

In 1992 and 1993, NIST released two fingerprint databases well suited for the development and testing of fingerprint classification systems: NIST Special Database 4 (Watson &

Wilson, 1992) and NIST Special Database 14 (Watson, 1993), hereinafter named DB4 and DB14, respectively. Both databases consist of 8-bit gray-level images of rolled fingerprint impressions scanned from cards; two different fingerprint instances (F and S) are present for each finger. Each fingerprint was manually analyzed by a human expert and assigned to one of the five classes: Arch (A), Left loop (L), Right loop (R), Tented arch (T), and Whorl (W). Actually, in DB4, some ambiguous fingerprints (about 17%) have an additional reference to a “secondary” class, and in DB14, there are a few fingerprints that the human expert was not able to classify. DB4 contains 2,000 fingerprint pairs, uniformly distributed in the five classes; the images are numbered from F0001 to F2000 and from S0001 to S2000. DB14 contains 27,000 fingerprint pairs whose class distribution resembles natural fingerprint distribution: the images are numbered from F00001 to F27000 and from S00001 to S27000. NIST DB4 and DB14 became de facto standard benchmarks for fingerprint classification, and most of the algorithms published in the last decade were tested on one of these databases.

Although DB4 and DB14 constitute very useful benchmarks for studies on fingerprint classification, they are not well suited for testing pre-selection approaches using live-scan images. In fact, online impressions rarely contain all the fingerprint singularities (usually they do not cover the entire fingerprint area) and this may cause problems for methods using a global description of fingerprints (e.g., orientation image).

Sections 5.3.1 and 5.3.2 of the previous version of this Handbook (Maltoni, 2009) report details on the performance of the main fingerprint classification approaches for which results on DB4 or DB14 are available. Concisely the best methods exhibit a classification error of about 4% for DB4 and 6% for DB14.

Galar et al. (2015b) report a comparison of the performance of the main approaches on NIST DB4 and three databases generated by SFinGe software tool (Cappelli et al., 2000a, b, 2004), by using two different metrics: accuracy rate and Choen’s kappa rate (Cohen, 1960) that evaluate the portion of hits that can be attributed to the classifier itself, relative to all the classifications that cannot be attributed to chance alone. However, in our opinion, SFinGe is not ideal for generating patterns to be used for benchmarking fingerprint classification, since the underlying local orientation model is quite simple and a classification technique could exploit the knowledge of this model to gain accuracy.

5.3.3 Search Strategies for Exclusive Classification

A classification technique alone is usually not sufficient: a retrieval strategy should also be defined according to the application requirements such as the desired accuracy and efficiency, the matching algorithm used to compare fingerprints, the presence of a human supervisor, and so on. In general, different pre-selection strategies may be defined: for instance, the search may be stopped when a fixed portion of the database has been explored, or as soon as a matching fingerprint is found (in AFIS, this requires the presence

of a human expert who visually examines the fingerprints that are considered sufficiently similar by the minutiae matcher and terminates the search when a true correspondence is found). If an exclusive classification technique is used, the following search strategies can be used:

- *Hypothesized class only*: only fingerprints belonging to the class to which the input fingerprint has been assigned are retrieved. The search may be stopped as soon as a matching fingerprint is found, or extended to all the fingerprints of that class in the database.
- *Fixed search order*: the search continues until a match is found, or the whole database has been explored; if a correspondence is not found within the hypothesized class, the search continues in another class, and so on. The optimal class visiting order can be a priori determined from the confusion matrix of a given fingerprint classifier (Lumini et al., 1997). For example, if the input fingerprint is assigned to the arch class, the order could be arch, tented arch, left loop, right loop, and whorl.
- *Variable search order*: the different classes are visited according to the class likelihoods produced by the classifier for the input fingerprint. The search may be stopped as soon as a match is found, or when the likelihood ratio between the current class and the next to be visited is less than a fixed threshold; see Senior (2001) or Senior and Bolle (2004).

Obviously, the first strategy (hypothesized class only) assumes no classification errors, which is quite unlikely for state-of-the-art automatic classifiers; the other strategies are more complex, but allow adjusting the accuracy of the system at the cost of speed. Each of the above three strategies may be combined with a rejection mechanism: if the input fingerprint is rejected by the automatic classifier, it has to be either manually classified or compared with all the fingerprints in the database.

5.4 Fingerprint Indexing and Retrieval

The main problem of the classification schemes discussed in the previous sections for indexing and retrieval (and of all the other commonly used schemes) is that the number of classes is small and fingerprints are unevenly distributed among them: more than 90% of the fingerprints belong to only three classes (right loop, left loop, and whorl). In 10-print identification (where an individual has to be identified using information from all of his 10 fingers), this does not compromise the efficiency too much, inasmuch as the knowledge of the classes of all the fingerprints can be used as a distinctive code for reducing the number of comparisons; on the other hand, when a single fingerprint has to be searched in a large database, the classification stage is not able to sufficiently narrow down the search. Furthermore, when classification is performed automatically, errors and rejected fingerprints are required to be handled gracefully. The intrinsic difficulties in automating fingerprint classification and sub-classification led some researchers to investigate fingerprint retrieval systems that are not based on human-defined classes. In fact, for applications where there is no need to adhere to Henry's classification scheme, and where the goal is purely to minimize the number of comparisons during fingerprint pre-selection, any technique able to characterize each fingerprint in a robust and stable manner (among different impressions of the same finger) may, in principle, be used.

Generally, indexing refers to the use of an appropriate representation and an efficient data structure that could be useful to reduce the matching costs, returning a list of candidates against which the query sample (probe) must be matched. Figure 5.10 shows a

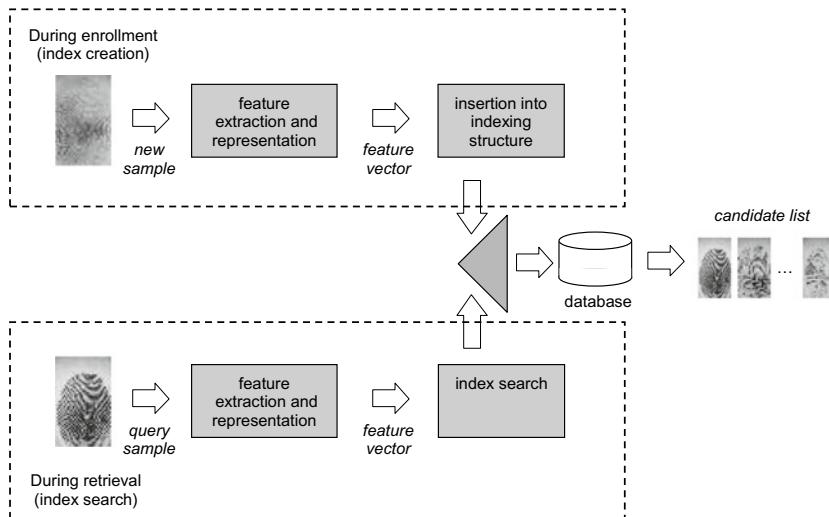


Fig. 5.10 Conceptual schema of indexing and retrieval

conceptual schema for indexing and retrieval.

There are two important aspects to be faced with indexing: the features used to represent relevant and distinctive characteristics of the fingerprints and the retrieval approach to be adopted. Most of the proposed solutions belong to one of the following families:

- *Indexing based on global features*: global features such as the local orientation image (or other Level-1 features) are used to derive a compact representation from a fingerprint image. Usually, the representation consists of a fixed-length vector, and an absolute pre-alignment of the fingerprint image is required to confer the necessary invariance. The feature vectors are created through a similarity-preserving transformation, so that similar fingerprints are mapped into close points (vectors) in the multidimensional space. Indexing/retrieval can be then performed by exhaustive search³ or by using spatial data structures (Samet, 1990) supporting neighboring searches. This approach was referred to as “*continuous classification*” by some authors (Lumini et al., 1997) to point out that fingerprints are not partitioned into disjoint classes, but associated with continuous numerical vectors.
- *Indexing based on local features*: indexing based on Level-2 features (such as minutiae) is very powerful but trickier because of the variable number and lack of order of such features. Descriptors derived from local arrangements of minutiae are used instead of single minutiae because of their invariance to geometric transformation. The inverted index approach, largely employed in information retrieval applications (Manning et al., 2008), is the most popular technique for the index creation and search.

Multiple indexes are often used together in the so-called *cascaded filtering* approach: the basic idea is to set up a sequential approach that progressively refines the search (Wilson et al., 2004; Jarosz et al., 2005). In principle, both global and local features may be adopted for filtering/indexing. Generally, the order of the filters is chosen such that the first filters are those with lower computational complexity and greater robustness, in spite of a limited selectivity; the ideal filters for the first steps of such a cascading strategy are able to quickly remove those fingerprints that significantly differ from the searched one, while avoiding to eliminate genuine candidates.

A recent survey of indexing approaches for biometric databases can be found in Gupta et al. (2019); the authors propose a taxonomy of indexing methods: texture-based, minutiae-based, hybrid, and Deep NN-based. A similar taxonomy is used across the following subsections to introduce the main relevant works.

³ If fixed-length compact vectors are used to represent fingerprints, the cost of exhaustive search is manageable since each comparison is a trivial distance computation (e.g., an Euclidean distance).

5.4.1 Methods Based on Orientation and Frequency Images

Most of the indexing techniques based on global features use the orientation image as an initial feature, but differ in the transformation adopted to create the final vectors and in the distance measure. In Lumini et al. (1997) and in Kamei and Mizoguchi (1998), the orientation image is aligned with respect to the core and treated as a single vector (by concatenating its rows); a dimensionality reduction (KL transform) is then performed to compute the final vectors. The similarity between two vectors is calculated with the Euclidean distance in Lumini et al. (1997) and with an approach that also takes into account a quality index, in Kamei and Mizoguchi (1998) (see also Kamei, 2004).

In some works, the orientation image is coupled with ridge-line frequency information: in Lee et al. (2005), a block-based local frequency is used, whereas in Jiang et al. (2006) and in Liu et al. (2007), a single value summarizing the dominant ridge-line frequency of the whole fingerprint is computed; in fact, Jiang, Liu, and Kot noted from their experiments that the local ridge-line frequency is much less stable than the local orientation, while a single scalar value encoding the dominant frequency is stable enough and has a good discriminant power. Finally, a more sophisticated approach is proposed by Wang et al. (2007): the authors use the coefficients of their FOMFE orientation model (see Sect. 3.3.5) as feature vectors for the continuous classification and report performance improvements with respect to feature vectors based on the raw orientation image.

In Cappelli et al. (1999a), the templates corresponding to the five classes (see Sect. 5.2) are used to create a numerical vector for continuous classification; the cost of the adaptation of each template to a given fingerprint is calculated; and a five-dimensional vector is assembled by using the five normalized costs. The main advantage of this approach is that the orientation images do not need to be aligned with respect to a fixed point. In Cappelli et al. (2000a), the vector adopted for continuous classification is created by using the distances of the orientation image from all the MKL subspaces.

The work by Cappelli (2011) introduces an indexing approach based on vector and scalar features, obtained from ridge-line orientations and frequencies. With a carefully designed set of features and ad hoc score measures, the indexing algorithm becomes very effective and efficient.

5.4.2 Methods Based on Matching Scores

Maeda et al. (2001) proposed a rather different approach, which is based only on the matching scores between fingerprint images rather than on features extracted from the fingerprint images: a matrix containing all the matching scores between each pair of fingerprints in the database is maintained. During retrieval, as the input fingerprint is matched with database fingerprints, the resulting scores are incrementally used to find the maximum correlation with a column of the matrix and to select the next database

fingerprint to be matched. The method is interesting, because it can be applied to any biometric identifier, inasmuch as it only relies on matching scores; on the other hand, it is not well suited for large databases (e.g., one million fingerprints) because the size of the matrix is quadratic with the number of fingerprints in the database, and for each new insertion in the database (enrollment), the new fingerprint has to be matched with all the fingerprints currently stored in the database. Becker and Potts (2007) propose another approach based only on matching scores, where a set of *fiduciary templates* are randomly selected in the database and the input fingerprint is compared against all those templates. The matching scores obtained are used as a feature vector for a continuous classification approach.

5.4.3 Methods Based on Minutiae

Several researchers proposed to index fingerprints using minutiae points: exploiting the same feature for matching and indexing fingerprints is attractive, but care must be taken to construct an extremely redundant representation, inasmuch as only a subset of all the minutiae is always present in different impressions of the same finger. The algorithm proposed by Germain et al. (1997) first identifies all the minutiae triplets in a fingerprint. Each triplet defines a triangle whose geometric features are extracted: length of each side, the angles, and the ridge count between each pair of vertices. The similarity between two fingerprints is defined by the number of corresponding minutiae triplets that can be found under a rigid transformation; this method of defining fingerprint similarity has strong analogies with local minutiae-based matching described in Sect. 4.4. Instead of explicitly comparing the similarity between the input fingerprint and all the fingerprints in the database (which would be very time-consuming), the authors use a geometric hashing technique: a hash table is built by quantizing all the possible triplets and, for each quantized triplet, a list of pointers (ID) to the fingerprints in the database containing that specific triplet is maintained. When a new fingerprint is inserted in the database, its triplets are extracted, and the hash table is updated by adding the fingerprint ID in the cell corresponding to the fingerprint triplets (Fig. 5.11). At retrieval time, the triplets of the input fingerprint are computed and quantized and, for each triplet, the list of fingerprint IDs in which that triplet is present is retrieved together with the coordinate transformations that best map the input fingerprint into the database fingerprints. Intuitively, if the same fingerprint ID is hit by more triplets in the input (under consistent coordinate transformations), then it is more likely that the corresponding fingerprint is the searched one. A voting technique is then applied to obtain a final ranking, which is used for visiting the database in a convenient order (Fig. 5.11).

Some variants of the above-described technique are presented in Bhanu and Tan (2001), Tan and Bhanu (2003), Bhanu and Tan (2003), and Choi et al. (2003), where more robust features are extracted from the minutiae triplets and geometric constraints are introduced

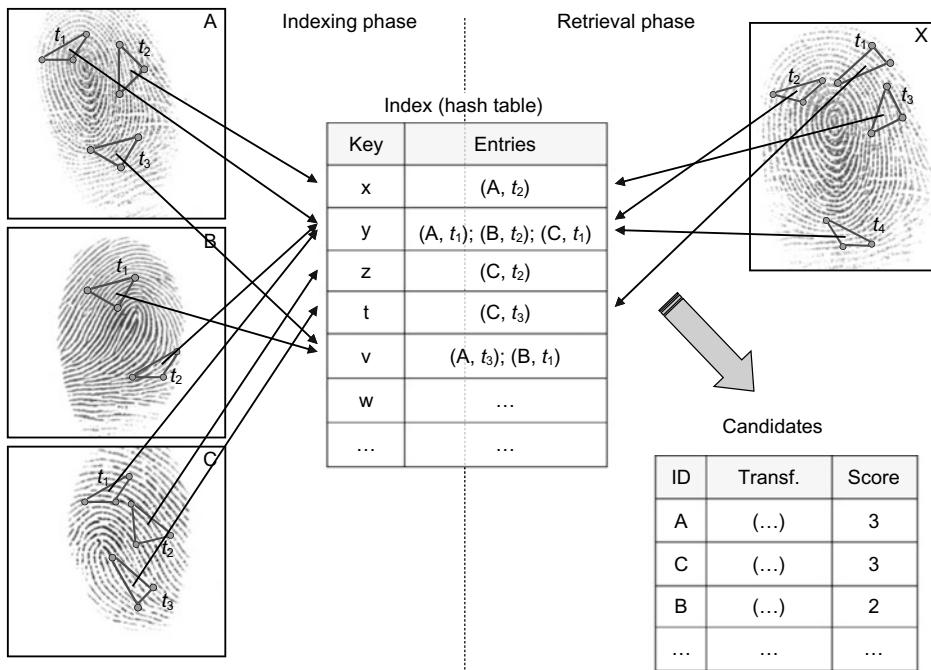


Fig. 5.11 An example of the approach proposed by Germain et al. (1997). During the indexing phase (on the left), the extracted features from each fingerprint in the database are used to generate the key values (x, y, z, ...) in the figure) for the hash table; for each key value, the index maintains the list of fingerprint IDs (A, B, C) and the corresponding triplets (t_1, t_2, \dots). During the retrieval phase, the triplets of the input fingerprint X are computed and the list of fingerprint IDs in which that triplet is present is retrieved, together with the coordinate transformations that best map X into the database fingerprints

to more effectively filter the database. Bebis et al. (1999) suggest the use of the Delaunay triangulation for selecting minutiae triplets: assuming an average number of m minutiae per fingerprint, this allows to consider only $O(m)$ minutiae triplets during indexing, thus saving memory and improving efficiency. The same technique is adopted in Ross and Mukherjee (2007), with the addition of information on ridge curvature to the minutiae triplets, resulting in an improved indexing performance. Liang et al. (2007) observe that the Delaunay triangulation is not tolerant enough to skin distortion that often affects fingerprints and shows how an algorithm that considers both order-0 and order-1 Delaunay triangles (see Gudmundsson et al. (2002)) can be more stable and robust against distortion.

Minutiae quadruplet features have been observed in Iloaunusi et al. (2011) and in Iloaunusi (2014) to be more robust than triplets. A minutia quadruplet is a quadrilateral composed of four minutiae points, and compared to minutiae triplets, they are claimed to be less sensitive to distortion. In Iloaunusi et al. (2011), features extracted from minutiae

quadruplets are clustered (using k-means) and the cluster centroids are used as atoms in a bag-of-words-based indexing approach. In spite of the higher accuracy reported in the experiments, this approach has higher computational complexity than triplet-based inverted index approaches.

The method proposed by Mansukhnai et al. (2010) arranges a fingerprint dataset into a tree-based index. Each intermediate node denotes an arrangement of minutiae points based on the path from the root to that node. Fingerprints are enrolled at the leaf nodes; one or more fingerprints can be enrolled at each leaf. Also, a fingerprint might be enrolled at multiple leaf nodes in the tree, representing multiple minutiae paths corresponding to different minutiae arrangements. During retrieval, the tree is traversed according to the minutiae in the probe fingerprint and the reached leaf nodes point the mated fingerprints in the database. Another tree-based indexing was proposed by Bai et al. (2015) where k-plet minutiae descriptors are used in combination with a multipath indexing strategy.

Cappelli et al. (2011) proposed an indexing method based on a Locality-Sensitive Hashing (LSH) scheme relying on Minutiae Cylinder-Code descriptors (see MCC in Sect. 4.4.3). Since MCC associates a robust fixed-length binary vector (i.e., descriptor) to each minutia, the use of a similarity-preserving hashing technique allows to map similar minutiae descriptors to the same entry in the hash table. In spite of the smaller set of features used (top-performing methods usually combine more features), in the experiments, the MCC-based indexing outperformed other existing approaches.

Another MCC-based indexing approach has been proposed in Su et al. (2016) where a global spatial transformation constraint is enforced. For this purpose, a learning-based fingerprint pose estimation algorithm is used to coarsely align fingerprints. This method also proposed an improved LSH algorithm for mapping the minutiae descriptors. Both fingerprint indexing accuracy and speed significantly improved in the experimental results; however, finger pose estimation remains a critical task for partial and poor-quality fingerprints as discussed in Sect. 3.5.5.

A finer hash bit selection method based on LSH and MCC is proposed in Zhou et al. (2021). It takes into consideration another feature, the single maximum collision, for indexing, and fuses the candidate lists produced by both indexing methods to produce the final candidate list. Experiments on multi-sensor databases (two-dimensional and three-dimensional databases) show that the proposed indexing approach improves the performance of fingerprint indexing.

Wang et al. (2015) proposed to learn compact binary hash codes for fingerprint indexing. This is done by applying the theory of Markov random field to model bit correlations in the translation-invariant MCC local descriptors, whose size is reduced from 384 to 24 bits. Further dimensionality reduction of MCC descriptors was introduced in Bai et al. (2018a, b), where feature transformations are learnt to reduce the descriptor dimensionality by minimizing inter-bit correlation and maximizing intra-bit variation. A multi-index hashing algorithm is then applied to extract k-nearest neighbors of the query fingerprint.

5.4.4 Hybrid and Ensemble Methods

In some papers, different indexing techniques are combined to improve performance. In Cappelli et al. (2000a), the continuous classification technique proposed in Cappelli et al. (1999a) is combined with an MKL-based approach: the distance measures produced by the two methods are fused by registering the different values according to their statistical distributions. In De Boer et al. (2001), two continuous classification techniques (the former similar to Lumini et al. (1997) and the latter using FingerCode feature vectors (Jain et al., 1999)) are combined with a simplified version of the minutiae-triplet approach proposed by Germain et al. (1997). In Sha and Tang (2004), continuous classification approaches are combined to exclusive classification techniques. In Cappelli and Ferrara (2012), a fingerprint retrieval system is introduced that combines: (i) local ridge-line orientations and frequencies (Level-1 features) and (ii) minutiae positions and angles (Level-2 features) represented by MCC binary vectors. Traditional rank-level (e.g., highest rank) and score-level (e.g., sum rule) fusion approaches have been evaluated and compared to the hybrid combination approach. In Paulino et al. (2013), a hybrid-indexing scheme for latent fingerprints has been experimented which uses singular points, frequency information, minutiae, and orientation field to build the index. The orientation field of each minutia is converted into a fixed-length bit vector which is translation and rotation invariant. LSH has been applied to index these bit vectors. Triplet indexing and orientation field indexing are combined with the MCC indexing approach. In Li et al. (2014), an indexing method using the features extracted from minutia details including location, direction, and ridge information is presented. A score-level fusion is proposed by combining the proposed method with MCC indexing. In Zhou et al. (2016), a hybrid approach that works for partial fingerprints has been proposed, applying separately a minutiae triplet-based indexing scheme and a two-dimensional Fourier expansion (FOMFE) coefficients-based indexing scheme to generate two candidate lists; then, a fuzzy-based fusion scheme is used to generate the final candidate list for matching. A hierarchical indexing scheme that combines the merits of LSH and geometric hashing permits to achieve a satisfactory identification accuracy being more robust in the presence of noise and missing points. A comprehensive list of various hybrid approaches proposed in the literature is reported in the survey by Gupta et al. (2019).

5.4.5 Deep Learning-Based Methods

The representation learning capabilities of deep neural networks trained on large datasets recently allowed to extract fixed-length representations which are very effective for fingerprint indexing and allowed to surpass previous techniques (see the comparison in Table 5.4). Cao and Jain (2017) argued that published indexing algorithms often do not meet the requirements, especially at low penetrate rates, because of the difficulty in extracting

Table 5.4 Comparison among various indexing approaches with reference to the most used databases for benchmarking. The first performance indicator ($\text{PR}@\text{HR} = 100\%$) shows the average penetration rate (PR) for the incremental search scenario; the other columns show the hit rate (HR) at various penetration rates

	Method	PR@HR = 100%	HR@			
			PR = 1%	PR = 5%	PR = 10%	PR = 20%
NIST DB4	Bhanu and Tan (2003)	–	–	–	85.5	–
	Jiang et al. (2006)	–	–	–	–	94.7
	Cappelli et al. (2011)	1.59	–	–	–	97.5
	Cappelli (2011)	1.47	–	–	–	–
	Cappelli and Ferrara (2012)	0.97	–	–	–	99.9
	Su et al. (2016)	–	–	–	97.03	–
	Bai et al. (2018a)	2.08	–	94.4	95.65	–
	Cao and Jain (2017)	–	98.65	99.25	99.60	–
	Song and Feng (2017)	0.79	–	–	–	–
	Bai et al. (2018b)	2.15	–	–	–	–
	Bai et al. (2018c)	1.95	–	–	–	–
	Song et al. (2019)	0.28	–	–	–	–
	Li et al. (2019)	–	99.82	–	–	–
	Engelsma et al. (2021)	–	99.75	–	–	–
NIST DB4 Natural	Lumini et al. (1997)	6.90	–	–	–	–
	Cappelli et al. (1999a)	5.22	–	–	–	–
	Lee et al. (2005)	3.85	–	–	–	–
	Jiang et al. (2006)	2.93	–	–	–	–

(continued)

Table 5.4 (continued)

	Method	PR@HR = 100%	HR@			
			PR = 1%	PR = 5%	PR = 10%	PR = 20%
	Cappelli et al. (2011)	1.32	—	—	—	—
	Cappelli (2011)	0.94	—	—	—	—
	Cappelli and Ferrara (2012)	0.68	—	—	—	—
	Song and Feng (2017)	0.67	—	—	—	—
	Song et al. (2019)	0.26	99.49	99.62	99.75	—
NIST DB14	Lumini et al. (1997)	7.11	—	—	—	—
	Cappelli et al. (1999a)	6.41	—	—	—	—
	Cappelli et al. (2002)	3.70	—	—	86.80	96.00
	Cappelli et al. (2011)	2.19	—	—	—	—
	Cappelli (2011)	1.80	—	—	—	—
	Cappelli and Ferrara (2012)	1.11	—	—	—	—
	Su et al. (2016)	—	—	—	97.67	—
	Bai et al. (2018a)	2.09	—	—	—	—
	Cao and Jain (2017)	—	98.93	99.74	99.81	—
	Bai et al. (2018b)	2.17	—	—	—	—
	Bai et al. (2018c)	1.97	—	—	—	—
	Song et al. (2019)	0.06	—	—	—	—
	Li et al. (2019)	—	99.91	—	—	—
	Engelsma et al. (2021)	—	99.93	—	—	—

(continued)

Table 5.4 (continued)

	Method	PR@HR = 100%	HR@			
			PR = 1%	PR = 5%	PR = 10%	PR = 20%
FVC2000 DB2	Cappelli et al. (2011)	1.72	—	—	—	—
	Cappelli (2011)	1.60	—	—	—	—
	Cappelli and Ferrara (2012)	1.17	—	—	—	—
	Zhou et al. (2016)	5.24	88.00	91.00	92.00	94.00
	Su et al. (2016)	—	—	—	98.51	—
	Bai et al. (2018a)	1.78	—	—	—	—
	Song and Feng (2017)	1.29	—	—	—	—
	Bai et al. (2018b)	2.28	—	—	—	—
	Bai et al. (2018c)	1.63	—	—	—	—
	Song et al. (2019)	1.02	—	—	—	—
FVC2000 DB3	Cappelli et al. (2011)	3.63	—	—	—	—
	Cappelli (2011)	3.49	—	—	—	—
	Cappelli and Ferrara (2012)	2.06	—	—	—	—
	Su et al. (2016)	—	—	—	94.14	—
	Bai et al. (2018a)	4.25	—	—	—	—
	Song and Feng (2017)	2.02	—	—	—	—
	Bai et al. (2018b)	4.89	—	—	—	—
	Bai et al. (2018c)	4.07	—	—	—	—
	Song et al. (2019)	1.21	—	—	—	—

(continued)

Table 5.4 (continued)

	Method	PR@HR = 100%	HR@			
			PR = 1%	PR = 5%	PR = 10%	PR = 20%
FVC2002 DB1	Cappelli et al. (2011)	1.37	–	–	–	–
	Cappelli (2011)	2.90	–	–	–	–
	Cappelli and Ferrara (2012)	1.26	–	–	–	–
	Zhou et al. (2016)	3.51	91.00	94.00	95.00	96.00
	Su et al. (2016)	–	–	–	99.29	–
	Bai et al. (2018a)	1.55	–	–	–	–
	Bai et al. (2018b)	1.56	–	–	–	–
	Bai et al. (2018c)	1.49	–	–	–	–

reliable minutiae and other features in low-quality fingerprint images. To overcome the limitation of minutiae-based indexing schemes, their work proposes a CNN-based fingerprint indexing algorithm. An orientation field dictionary is learned to align fingerprints in a unified coordinate system, and a large longitudinal fingerprint database (440 K fingerprints), where each finger has multiple impressions over time, is used to train the CNN. It is worth noting that the availability of multiple impressions for the same finger is essential because the network is trained to group fingerprints by finger. The output of the last fully connected layer of the CNN is used as the 2048-dimensional fixed-length feature vector for indexing.

Song and Feng (2017) used CNNs to extract multilevel features (Levels 1, 2, and 3) by cropping fingerprint portions at three different scales. These features are aggregated into a fixed-length vector denoted as Pyramid Deep Convolutional (PDC) feature (see Fig. 5.12). Here too, a preliminary registration according to the core point is performed to confer translation invariance.

In the method by Song et al. (2019), minutia-centered local descriptors are learned by a first CNN and then aggregated into a fixed-length descriptor by a second CNN (AggregationNet). The AggregationNet: (i) embeds each local descriptor into a high-dimensional space through one-dimensional convolutional layers; (ii) merges the set of embedded vectors into a single fixed vector by average pooling; and (iii) is trained by metric learning to ensure that the aggregated features are discriminative and compact. The main advantage of this technique is that it is not requiring any pre-alignment which is known to be a critical step. On the other hand, it requires to extract minutiae with classical approaches.

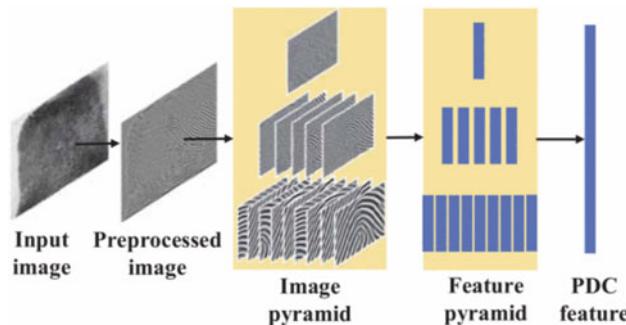


Fig. 5.12 The Pyramidal deep convolutional feature proposed by Song and Feng (2017). © IEEE. Reprinted, with permission, from Song and Feng (2017)

An “implicit” patch aggregation method was proposed by Li et al. (2019) where neither fingerprint alignment nor minutiae extraction is necessary at retrieval time. A fully convolutional neural network (i.e., an architecture without fully connected layers and with a single final global pooling layer) is trained to group fingerprint patches by triplet loss while producing a fixed-length encoding of 256 bytes. Because of the fully convolutional architecture, the trained model can be used to aggregate the whole content of a full fingerprint image in a vector of the same size.

In Bai et al. (2018c), a deep compact binary minutia cylinder code (DCBMCC) is proposed as an effective and discriminative feature representation for fingerprint indexing. The CNN used to learn the DCBMCC representation restricts the penultimate layer to directly output binary codes. Moreover, independence, balance, quantization-loss-minimum, and similarity-preservation properties are considered during the learning process. A multi-index hashing-based fingerprint indexing scheme speeds up the exact search in the Hamming space by building multiple hash tables on binary code substrings.

In the work by Engelsma et al. (2021), a CNN, named DeepPrint, is adopted to learn to extract fixed-length fingerprint representations of 200 bytes. Even if this approach includes fingerprint alignment and minutiae detection as intermediate steps, they are performed by subnets embedded in a global model which is trained end to end (see Sect. 4.6.5 for more details). The compact DeepPrint representation can significantly speed up large-scale fingerprint search. This approach, compared with two state-of-the-art COTS matchers on a gallery of 1.1 million fingerprints, shows competitive search accuracy at significantly faster speeds.

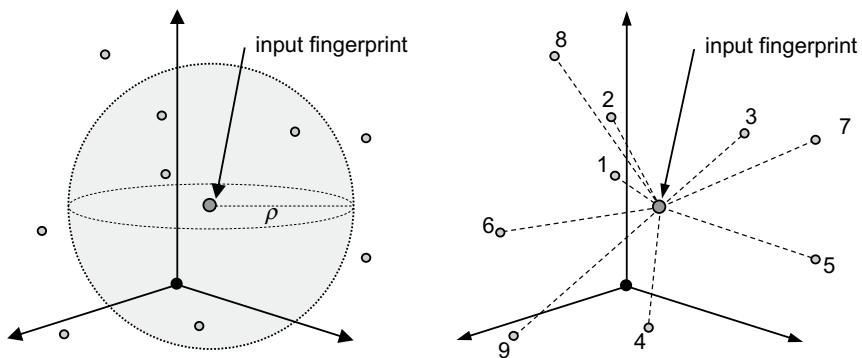


Fig. 5.13 Retrieval strategies for indexing based on global features (a.k.a. continuous classification). On the left: a hypersphere is considered with center on the query and radius determined by the penetration rate (*fixed value of PR*). On the right: the fingerprints are incrementally retrieved according to the distance of the corresponding vectors from the input point (*incremental search*)

5.5 Benchmarking Fingerprint Indexing Techniques

5.5.1 Metrics and Benchmarks

Fingerprint indexing approaches are commonly evaluated by considering the *hit rate* (HR) at a given *penetration rate* (PR), where the former denotes the probability of finding the searched fingerprint in the candidate list and the latter the average database portion to be explored. Two retrieval strategies can be considered:

- *Fixed penetration*: the search is halted as soon as a match is found, or when a given maximum portion of the database (PR) has been explored. For example, we can measure $\text{HR@PR} = 5\%$.
- *Incremental search*: fingerprints are visited according to their similarity to the searched fingerprint (*query*); the search continues until a match is found (in the worst case, it is extended to the whole database). In such a scenario, there are no retrieval errors, since in the worst case the search can be extended to the whole database. An advantage of this approach is that the performance of an algorithm over a dataset can be summarized with just a single value (i.e., $\text{PR@HR} = 100\%$).

In the case of indexing based on global features, the feature vectors lie in a multidimensional space, and a simple geometric interpretation can be associated with the two search strategies (see Fig. 5.13).

FVC-onGoing platform includes two benchmarks for fingerprint indexing approaches⁴:

⁴ <https://biolab.csr.unibo.it/fvcongoing/UI/Form/BenchmarkAreas/BenchmarkAreaFIDX.aspx>.

- FIDX-10K-1.0: consists of ten thousand fingerprints to be indexed and one hundred fingerprints to be searched.
- FIDX-50K-1.0: consists of fifty thousand fingerprints to be indexed and five hundred fingerprints to be searched.

In both the cases, each query has one mate in the indexed ones. Images were acquired in operational conditions using high-quality optical scanners.

The performance of evaluated approaches is reported as a trade-off between error rate and penetration rate. Constantly updated results are available in the dedicated section on the FVC-onGoing website.⁵

5.5.2 Comparison of Existing Approaches

This section summarizes a comparative evaluation of indexing schemes discussed so far. It is not a simple task to provide performance comparisons considering that not all methods use the same protocol or conduct experiments on the same databases. Due to these problems, the comparison in Table 5.4 is limited to some methods that adopt the same retrieval scenarios and/or exhibit results for the same fingerprint database. In general, we can observe the methods combining local and global features and different indexing schemes perform the best; moreover, the evolution of deep neural networks has certainly provided state-of-the-art results even for difficult databases.

5.6 Summary

Fingerprint classification has been the subject of several pattern recognition studies over the last five decades. Different solutions have been proposed and it is now possible to design classification systems that are able to meet the FBI requirement of 99% accuracy with a maximum rejection of 20%. However, it is unlikely that exclusive classification would make it possible to significantly reduce the effort of searching for a single fingerprint in the absence of other information (meta data) of the associated subject (e.g., sex, age, race, etc.).

Pre-selection and indexing strategies based on features extracted from fingerprints constitute better alternatives for efficient implementations of the search (identification) task in a variety of applications. In the last three decades, several fingerprint indexing techniques have been designed based on both local and global features. Hashing techniques based on local descriptors (e.g., minutiae triplets or MCC) have been the primary approach for several years, but recently they have been supplanted by fixed-length global representation

⁵ <https://biolab.csr.unibo.it/fvcongoing/UI/Form/PublishedAlgs.aspx>.

extracted by deep learning-based approaches. The best performing methods now allow to search a database with a hit rate of over 99% at a penetration rate = 1%.

While this chapter did not address the search speed-up made possible by parallel computation, before designing complex pre-selection and indexing techniques, one should pragmatically consider that efficient implementation of matching techniques can nowadays perform more than 100 million fingerprint matching per second on a single PC with a GPU (see Sect. 4.4.3).

References

- Bai, C., Li, M., Zhao, T., & Wang, W. (2018a). Learning binary descriptors for fingerprint indexing. *IEEE Access*, 6, 1583–1594.
- Bai, C., Wang, W., Zhao, T., & Li, M. (2018b). Fast exact fingerprint indexing based on compact binary minutia cylinder codes. *Neurocomputing*, 275, 1711–1724.
- Bai, C., Wang, W., Zhao, T., Wang, R., & Li, M. (2018c). Deep learning compact binary codes for fingerprint indexing. *Frontiers of Information Technology & Electronic Engineering*, 19, 1112–1123.
- Bai, C., Zhao, T., Wang, W., & Wu, M. (2015). An efficient indexing scheme based on K-plet representation for fingerprint database. In *Proceedings International Conference on Intelligent Computing*.
- Ballan, M., Sakarya, F. A., & Evans, B. L. (1997). A fingerprint classification technique using directional images. In *Proceedings of Asilomar Conference on Signals Systems and Computers*.
- Bartesaghi, A., Fernández, A., & Gómez, A. (2001). Performance evaluation of an automatic fingerprint classification algorithm adapted to a Vucetich based classification system. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 259–265).
- Bebis, G., Deaconu, T., & Georgopoulos, M. (1999). Fingerprint identification using delaunay triangulation. In *Proceedings of IEEE International Conference on Intelligence, Information, and Systems* (pp. 452–459).
- Becker, G., & Potts, M. (2007). Non-metric biometric clustering. In *Proceedings of Biometric Symposium*.
- Bernard, S., Boujemaa, N., Vitale, D., & Bricot, C. (2001). Fingerprint classification using kohonen topologic map. In *Proceedings of International Conference on Image Processing*.
- Bhanu, B., & Tan, X. (2001). A triplet based approach for indexing of fingerprint database for identification. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 205–210).
- Bhanu, B., & Tan, X. (2003). Fingerprint indexing based on novel features of minutiae triplets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5), 616–622.
- Bowen, J. (1992). The home office automatic fingerprint pattern classification project. In *Proceedings of IEE Colloquium on Neural Networks for Image Processing Applications*.
- Bunke, H. (1993). Structural and syntactic pattern recognition. In C. H. Chen et al. (Eds.), *Handbook of pattern recognition & computer vision*. World Scientific.
- Candela, G. T., Grother, P. J., Watson, C. I., Wilkinson, R. A., & Wilson, C. L. (1995, August). PCASYS—A pattern-level classification automation system for fingerprints (Tech. Report: NIST TR 5647).

- Cao, K., & Jain, A. K. (2017). Fingerprint indexing and matching: An integrated approach. In *Proceedings of International Joint Conference on Biometrics*.
- Cao, K., Pang, L., Liang, J., & Tian, J. (2013). Fingerprint classification by a hierarchical classifier. *Pattern Recognition*, 46(12), 3186–3197.
- Cappelli, R., Lumini, A., Maio, D., & Maltoni, D. (1999a). Fingerprint classification by directional image partitioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5), 402–421.
- Cappelli, R., Maio, D., & Maltoni D. (1999b). Fingerprint classification based on multi-space KL. In *Proceedings of Workshop on Automatic Identification Advances Technologies* (pp. 117–120).
- Cappelli, R., Maio, D., & Maltoni, D. (2000a). Combining fingerprint classifiers. In *1st Proceedings of International Workshop on Multiple Classifier Systems* (pp. 351–361).
- Cappelli, R., Maio, D., & Maltoni, D. (2000b). Synthetic fingerprint-image generation. In *15th Proceedings of International Conference on Pattern Recognition* (Vol. 3, pp. 475–478).
- Cappelli, R., Maio, D., & Maltoni D. (2001). Multi-space KL for pattern representation and classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9), 977–996.
- Cappelli, R., Maio, D., & Maltoni, D. (2002). A multi-classifier approach to fingerprint classification. *Pattern Analysis and Applications (Special Issue on Fusion of Multiple Classifiers)*, 5(2), 136–144.
- Cappelli, R., Maio, D., Maltoni, D., & Nanni, L. (2003). A two-stage fingerprint classification system. In *Proceedings of ACM SIGMM Multimedia Biometrics Methods and Applications Workshop* (pp. 95–99).
- Cappelli, R., & Maio D. (2004). State-of-the-art in fingerprint classification. In N. Ratha & R. Bolle (Eds.), *Automatic fingerprint recognition systems* (pp. 183–205). Springer.
- Cappelli, R., Maio, D., & Maltoni, D. (2004). An improved noise model for the generation of synthetic fingerprints. In *Proceedings of International Conference on Control, Automation, Robotics and Vision*.
- Cappelli, R. (2011). Fast and accurate fingerprint indexing based on ridge orientation and frequency. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 41(6), 1511–1521.
- Cappelli, R., Ferrara, M., & Maltoni, D. (2011). Fingerprint indexing based on minutia cylinder-code. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5).
- Cappelli, R., & Ferrara, M. (2012). A fingerprint retrieval system based on level-1 and level-2 features. *Expert Systems with Applications*, 39(12), 10465–10478.
- Chang, J. H., & Fan, K. C. (2002). A new model for fingerprint classification by ridge distribution sequences. *Pattern Recognition*, 35(6), 1209–1223.
- Cho, B. H., Kim, J. S., Bae, J. H., Bae, I. G., & Yoo, K. Y. (2000). Core-based fingerprint image classification. In *Proceedings of International Conference on Pattern Recognition (15th)* (Vol. 2, pp. 863–866).
- Choi, K., Lee, D., Lee, S., & Kim, J. (2003). An improved fingerprint indexing algorithm based on the triplet approach. In *4th Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 584–591).
- Chong, M. M. S., Ngee, T. H., Jun, L., & Gay, R. K. L. (1997). Geometric framework for fingerprint image classification. *Pattern Recognition*, 30(9), 1475–1488.
- Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1), 37–46.
- De Boer, J., Bazen, A. M., & Gerez, S. H. (2001). Indexing fingerprint databases based on multiple features. In *Proceedings of Workshop on Circuits Systems and Signal Processing (ProRISC 2001)*.
- Drets, G., & Liljenstrom, H. (1998). Fingerprint sub-classification and singular point detection. *International Journal of Pattern Recognition and Artificial Intelligence*, 12(4), 407–422.

- Engelsma, J. J., Cao, K., & Jain, A. K. (2021). Learning a fixed-length fingerprint representation. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 43(6), 1981–1997.
- Federal Bureau of Investigation. (1984). *The science of fingerprints: Classification and uses*. U.S. Government Publication.
- Fitz, A. P., & Green, R. J. (1996). Fingerprint classification using hexagonal fast Fourier transform. *Pattern Recognition*, 29(10), 1587–1597.
- Fu, K. S., & Booth, T. L. (1986a). Grammatical inference: Introduction and survey: Part I. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(3), 343–360.
- Fu, K. S., & Booth, T. L. (1986b). Grammatical inference: Introduction and survey: Part II. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(3), 360–376.
- Galar, M., Derrac, J., Peralta, D., Triguero, I., Paternain, D., Lopez-Molina, C., García, S., Benítez, J. M., Pagola, M., Barrenechea, E., Bustince, H., Herrera, F. (2015a). A survey of fingerprint classification Part I: Taxonomies on feature extraction methods and learning models. *Knowledge-Based Systems*, 81, 76–97.
- Galar, M., Derrac, J., Peralta, D., Triguero, I., Paternain, D., Lopez-Molina, C., García, S., Benítez, J. M., Pagola, M., Barrenechea, E., Bustince, H., Herrera, F. (2015b). A survey of fingerprint classification Part II: Experimental analysis and ensemble proposal. *Knowledge-Based Systems*, 81, 98–116.
- Galton, F. (1892). *Finger prints*. Macmillan.
- Germain, R., Califano, A., & Colville, S. (1997). Fingerprint matching using transformation parameter clustering. *IEEE Computational Science and Engineering*, 4(4), 42–49.
- Grasselli, A. (1969). On the automatic classification of fingerprints. In S. Watanabe (Ed.), *Methodologies of pattern recognition*. Academic.
- Gudmundsson, J., Hammar, M. H., & Van Kreveld, M. (2002). Higher order delaunay triangulations. *Computer Geometry Theory Application*, 23(1), 85–98.
- Gupta, P., Tiwari, K., & Arora, G. (2019). Fingerprint indexing schemes—A survey. *Neurocomputing*, 335, 352–365.
- Halici, U., & Ongun, G. (1996). Fingerprint classification through self-organizing feature maps modified to treat uncertainties. *Proceedings of the IEEE*, 84(10), 1497–1512.
- Henry, E. (1900). *Classification and uses of finger prints*. Routledge.
- Hong, L., & Jain, A. K. (1999). Classification of fingerprint images. In *11th Proceedings of Scandinavian Conference on Image Analysis*.
- Hong, J. H., Min, J. K., Cho, U. K., & Cho, S. B. (2008). Fingerprint classification using one-vs-all support vector machines dynamically ordered with naive Bayes classifiers. *Pattern Recognition*, 41(2), 662–671.
- Hughes, P., & Green, A. (1991). The use of neural networks for fingerprint classification. In *2nd Proceedings of International Conference on Neural Networks*.
- Iloanusi, O. (2014). Fusion of finger types for fingerprint indexing using minutiae quadruplets. *Pattern Recognition Letters*, 38(1), 8–14.
- Iloanusi, O., Gyaourova, A., & Ross, A. (2011). Indexing fingerprints using minutiae quadruplets. In *Proceedings Computer Vision Pattern Recognition Workshops* (pp. 127–133). Colorado Springs, CO.
- Jain, A. K., Prabhakar, S., & Hong, L. (1999). A multichannel approach to fingerprint classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4), 348–359.
- Jain, A. K., Duin, P. W., & Mao, J. (2000). Statistical pattern recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1), 4–37.
- Jain, A. K., & Minut, S. (2002). Hierarchical kernel fitting for fingerprint classification and alignment. In *16th Proceedings of International Conference on Pattern Recognition*.

- Jarosz, H., Founder, J. C., & Dupre, X. (2005). Large-scale identification system design. In J. Wayman et al. (Eds.), *Biometric systems: Technology, design and performance evaluation*. Springer.
- Jian, W., Zhou, Y., & Liu, H. (2020). Lightweight convolutional neural network based on singularity ROI for fingerprint classification. *IEEE Access*, 8, 54554–54563.
- Jiang, X., Liu, M., & Kot, A. C. (2006). Fingerprint retrieval for identification. *IEEE Transactions on Information Forensics and Security*, 1(4), 532–542.
- Jolliffe, I. T. (1986). *Principle component analysis*. Springer.
- Kamei, T. (2004). Fingerprint preselection using eigenfeatures for large-size database. In N. Ratha & R. Bolle (Eds.), *Automatic fingerprint recognition systems* (pp. 263–282). Springer.
- Kamei, T., & Mizoguchi, M. (1998). Fingerprint preselection using eigenfeatures. In *Proceedings of Conference Computer Vision and Pattern Recognition* (pp. 918–923).
- Kamijo, M. (1993). Classifying fingerprint images using neural network: Deriving the classification state. In *Proceedings of International Conference on Neural Networks*.
- Kamijo, M., Mieno, H., & Kojima K., (1992). Classification of fingerprint images using a neural network. *Systems and Computers in Japan*, 23, 89–101.
- Karu, K., & Jain, A. K. (1996). Fingerprint classification. *Pattern Recognition*, 29(3), 389–404.
- Kawagoe, M., & Tojo, A. (1984). Fingerprint pattern classification. *Pattern Recognition*, 17(3), 295–303.
- Klimanee, C., & Nguyen, D. T. (2004). Classification of fingerprints using singular points and their principal axes. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 849–852).
- Kristensen, T., Borthen, J., & Fyllingsnes, K. (2007). Comparison of neural network based fingerprint classification techniques. In *Proceedings of International Joint Conference on Neural Networks* (pp. 1043–1048).
- Lee, S. O., Kim, Y. G., & Park, G. T. (2005). A feature map consisting of orientation and inter-ridge spacing for fingerprint retrieval. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication (5th)* (pp. 184–190).
- Li, J., Yau, W. Y., & Wang, H. (2008). Combining singular points and orientation image information for fingerprint classification. *Pattern Recognition*, 41(1), 353–366.
- Li, G., Yang, B., & Busch, C. (2014). A score-level fusion fingerprint indexing approach based on minutiae vicinity and minutia cylinder-code. In *2nd International Workshop on Biometrics and Forensics* (pp. 1–6).
- Li, R., Song, D., Liu, Y., & Feng, J. (2019). Learning global fingerprint features by training a fully convolutional network with local patches. In *Proceedings of International Conference on Biometrics (ICB)*, Crete, Greece.
- Liang, X., Bishnu, A., & Asano, T. (2007). A robust fingerprint indexing scheme using minutia neighborhood structure and low-order delaunay triangles. *IEEE Transactions on Information Forensics and Security*, 2(4), 721–733.
- Liu, M. (2010). Fingerprint classification based on Adaboost learning from singularity features. *Pattern Recognition*, 43(3), 1062–1070.
- Liu, M., Jiang, X., & Kot, A. C. (2007). Efficient fingerprint search based on database clustering. *Pattern Recognition*, 40(6), 1793–1803.
- Lumini, A., Maio, D., & Maltoni, D. (1997). Continuous vs exclusive classification for fingerprint retrieval. *Pattern Recognition Letters*, 18(10), 1027–1034.
- Lumini, A., Maio, D., & Maltoni, D. (1999). Inexact graph matching for fingerprint classification. *Machine Graphics & Vision (Special Issue on Graph Transformations in Pattern Generation and CAD)*, 8(2), 231–248.

- Maeda, T., Matsushita, M., & Sasakawa, K. (2001). Identification algorithm using a matching score matrix. *IEICE Transactions on Information and Systems (Special Issue on Biometrics)*, E84-D(7), 819–824.
- Maio, D., & Maltoni, D. (1996). A structural approach to fingerprint classification. In *13th Proceedings International Conference on Pattern Recognition*.
- Maltoni, D., Maio, D., Jain, A. K., & Prabhakar, S. (2009). *Handbook of fingerprint recognition* (2nd ed). Springer.
- Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge University Press.
- Mansukhani, P., Tulyakov, S., & Govindaraju, V. (2010). A framework for efficient fingerprint identification using a minutiae tree. *IEEE Systems Journal*, 4(2), 126–137.
- Marcialis, G. L., Roli, F., & Frasconi, P. (2001). Fingerprint classification by combination of flat and structural approaches. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication (3rd)* (pp. 241–246).
- Min, J. K., Hong, J. H., & Cho, S. B. (2006). Effective fingerprint classification by localized models of support vector machines. In *Proceedings International Conference on Biometrics*. LNCS (Vol. 3832, pp. 287–293).
- Moayer, B., & Fu, K. (1973). A syntactic approach to fingerprint pattern recognition. In *Proceedings of International Joint Conference on Pattern Recognition*.
- Moayer, B., & Fu, K. (1975). A syntactic approach to fingerprint pattern recognition. *Pattern Recognition*, 7(1–2), 1–23.
- Moayer, B., & Fu, K. (1976). An application of stochastic languages to fingerprint pattern recognition. *Pattern Recognition*, 8(3), 173–179.
- Moayer, B., & Fu, K. (1986). A tree system approach for fingerprint pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(3), 376–388.
- Moenssens, A. (1971). *Fingerprint techniques*, Chilton Book Company.
- Moscinska, K., & Tyma, G. (1993). Neural network based fingerprint classification. In *3rd Proceedings of International Conference on Artificial Neural Networks*.
- Neuhaus, M., & Bunke, H. (2005). A graph matching based approach to fingerprint classification using directional variance. In *5th Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 191–200).
- Omidvar, O. M., Blue, J. L., & Wilson, C. L. (1995). Improving neural network performance for character and fingerprint classification by altering network dynamics. In *Proceedings of World Congress on Neural Networks*.
- Pal, S. K., & Mitra, S. (1996). Noisy fingerprint classification using multilayer perceptron with fuzzy geometrical and textural features. *Fuzzy Sets and Systems*, 80(2), 121–132.
- Park, C. H., & Park, H. (2005). Fingerprint classification using fast Fourier transform and nonlinear discriminant analysis. *Pattern Recognition*, 38(4), 495–503.
- Pattichis, M. S., Panayi, G., Bovik, A. C., & Hsu, S. P. (2001). Fingerprint classification using an AM–FM model. *IEEE Transactions on Image Processing*, 10(6), 951–954.
- Paulino, A. A., Liu, E., Cao, K., & Jain, A. K. (2013). Latent fingerprint indexing: Fusion of level 1 and level 2 features. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems (BTAS)* (pp. 1–8).
- Peralta, D., Triguero, I., García, S., Saeys, Y., Benítez, J. M., & Herrera, F. (2018). On the use of convolutional neural networks for robust classification of multiple fingerprint captures. *International Journal of Intelligent Systems*, 33(1), 213–230.
- Rao, K., & Balck, K. (1980). Type classification of fingerprints: A syntactic approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(3), 223–231.

- Ratha, N. K., Karu, K., Chen, S., & Jain, A. K. (1996). A real-time matching system for large fingerprint databases. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8), 799–813.
- Ross, A., & Mukherjee, R. (2007). Augmenting ridge curves with minutiae triplets for fingerprint indexing. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification IV*.
- Samet, H. (1990). *The design and analysis of spatial data structures*. Addison-Wesley.
- Senior, A. (1997). A hidden markov model fingerprint classifier. In *31st Proceedings of Asilomar Conference on Signals Systems and Computers* (pp. 306–310).
- Senior, A. (2001). A combination fingerprint classifier. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10), 1165–1174.
- Senior, A. W., & Bolle, R. (2004). Fingerprint classification by decision fusion. In N. Ratha & R. Bolle (Eds.), *Automatic fingerprint recognition systems* (pp. 207–227). Springer.
- Sha, L., & Tang, X. (2004). Combining exclusive and continuous fingerprint classification. In *Proceedings of International Conference on Image Processing*, 2, 1245–1248.
- Shah, S., & Sastry, P. S. (2004). Fingerprint classification using a feedback-based line detector. *IEEE Transaction on Systems, Man, and Cybernetics, Part B*, 34(1), 85–94.
- Song, D., & Feng, J. (2017). Fingerprint indexing based on pyramid deep convolutional feature. In *Proceedings of International Joint Conference on Biometrics (IJCB)* (pp. 200–207).
- Song, D., Tang, Y., & Feng, J. (2019). Aggregating minutia-centred deep convolutional features for fingerprint indexing. *Pattern Recognition*, 88, 397–408.
- Su, Y., Feng, J., & Zhou, J. (2016). Fingerprint indexing with pose constraint. *Pattern Recognition*, 54, 1–13.
- Tan, X., & Bhanu, B. (2003). A robust two step approach for fingerprint identification. *Pattern Recognition Letters*, 24(13), 2127–2134.
- Tan, X., Bhanu, B., & Lin, Y. (2005). Fingerprint classification based on learned features. *IEEE Transaction on Systems, Man, and Cybernetics, Part C*, 35(3), 287–300.
- Tertychnyi, P., Ozcinar, C., & Anbarjafari, G. (2018). Low-quality fingerprint classification using deep neural network. *IET Biometrics*, 7(6), 550–556.
- Tou, J. T., & Hankley, W. J. (1968). Automatic fingerprint interpretation and classification via contextual analysys and topological coding. In C. Cheng, S. Ledley, D. Pollock, & A. Rosenfeld (Eds.), *Pictorial pattern recognition* (pp. 411–456). Thompson Book.
- Verma, M. R., & Chatterjee, B. (1989). Partial fingerprint pattern classification. *Journal Institute Electronic and Telecommunication Engineers*, 3(1), 28–33.
- Wang, L., & Dai, M. (2007). Application of a new type of singular points in fingerprint classification. *Pattern Recognition Letters*, 28(13), 1640–1650.
- Wang, Y., Hu, J., & Phillips, D. (2007). A fingerprint orientation model based on 2D Fourier expansion (FOMFE) and its application to singular-point detection and fingerprint indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4), 573–585.
- Wang, Y., Wang, L., Cheung, Y. M., & Yuen, P. C. (2015). Learning compact binary codes for hash-based fingerprint indexing. *IEEE Transactions on Information Forensics and Security*, 10(8), 1603–1616.
- Wang, X., & Xie, M. (2004). Fingerprint classification: An approach based on singularities and analysis of fingerprint structure. In *1st Proceedings of International Conference on Biometric Authentication. LNCS* (Vol. 3072, pp. 324–329).
- Watson, C.I. (1993). *NIST Special Database 14, Fingerprint Database*. U.S. National Institute of Standards and Technology.
- Watson, C. I., & Wilson, C. L. (1992). *NIST Special Database 4, Fingerprint Database*. U.S. National Institute of Standards and Technology.

- Wilson, C. L., Candela, G. T., & Watson, C. I. (1994). Neural network fingerprint classification. *Journal of Artificial Neural Networks*, 1(2), 203–228.
- Wilson, C.L., Garris, M. D., & Watson, C. I. (2004). *Matching performance for the US–VISIT IDENT system using flat fingerprints* (NIST Research Report: NISTIR 7110).
- Yao, Y., Marcialis, G. L., Pontil, M., Frasconi, P., & Roli, F. (2003). Combining flat and structured representations for fingerprint classification with recursive neural networks and support vector machines. *Pattern Recognition*, 36(2), 397–406.
- Yong, A., Guo, T., Wu, Y., & Shao, G. (2013). Fingerprint subclassification using rotation-invariant features. In *Proceedings of International Conference on Intelligent Control and Information Processing (ICICIP)*, Beijing, China.
- Yuan, Q., Tian, J., & Dai, R. (1998). Fingerprint classification system with feedback mechanism based on genetic algorithm. In *14th Proceedings of International Conference on Pattern Recognition*.
- Zhang, Q., & Yan, H. (2004). Fingerprint classification based on extraction and analysis of singularities and pseudo ridges. *Pattern Recognition*, 37(11), 2233–2243.
- Zhou, W., Hu, J., & Wang, S. (2021). Enhanced locality-sensitive hashing for fingerprint forensics over large multi-sensor databases. *IEEE Transactions on Big Data*, 7(4), 759–769.
- Zhou, W., Hu, J., Wang, S., Petersen, I., & Bennamoun, M. (2016). Partial fingerprint indexing: A combination of local and reconstructed global features. *Concurrency and Computation Practice and Experience*, 28(10), 2940–2957.



Latent Fingerprint Recognition

6

Abstract

Latent fingerprints (or fingermarks) are the traces of friction ridges left by fingers touching object surfaces. Because of their poor quality, latent fingerprints have been (and still often are) processed semi-automatically; a latent print examiner marks the minutiae in latent prints, followed by an automatic search on an AFIS. This procedure is time-consuming and subjective, which limits both the efficiency and efficacy of identifying the suspect. After an introduction to the procedure used by forensic latent examiners for latent fingerprint recognition, this chapter reviews the main techniques enabling fully automatic (or lights-out operation) processing of latent fingerprints, focusing on recent deep learning-based feature extraction and matching approaches. Specific sections on latent fingerprint quality and performance evaluation are finally provided.

Keywords

Latent fingerprints • Fingermarks • ACE-V procedure • Latent examiners • Lights-out processing • Latent fingerprint quality

6.1 Introduction

A latent fingerprint (or fingermark) is an impression that is made when a finger touches an object. The quality (i.e., contrast, size, distortion) of such an impression is determined by the interaction between a finger and the object's surface. The use of latent fingerprints to identify suspects of crime has a long history. Prior to the advent of AFIS, latent fingerprint identification by latent examiners was feasible only if the suspect was already in custody, since there was no practical indexing technique for searching a latent fingerprint in a large

database of exemplar (or reference) fingerprints. With the advent of AFIS, blind searches without any apprehended suspects became possible. Today, latent fingerprint recognition technology is one of the most powerful tools available to law enforcement agencies all over the world. In 2020 alone, the FBI's Next Generation Identification (NGI) System received close to three hundred thousand requests for latent fingerprint search from all over the United States (FBI, 2021).

Compared to fingerprints obtained by live-scan devices and scanned finger cards, latent fingerprints are very challenging for AFIS mainly due to three factors: poor image quality, small finger area, and large deformation in the print. When these factors are simultaneously present, latent fingerprint identification is particularly difficult. The fact that latent fingerprint recognition usually works in identification mode (1:N search) for a large database makes the latent search even more challenging. Figure 6.1 shows three latent fingerprints along with their mated rolled fingerprints.

Due to the above difficulties, the recognition accuracy of latent fingerprints is much lower than that of plain or rolled fingerprints. In NIST's evaluation of rolled and plain fingerprint recognition technology, FpVTE 2012 (Watson et al., 2012), the best performing AFIS achieved a false negative identification rate (FNIR) of 1.9% for single index fingers, at a false positive identification rate (FPIR) of 0.1% using 30,000 search subjects (10,000

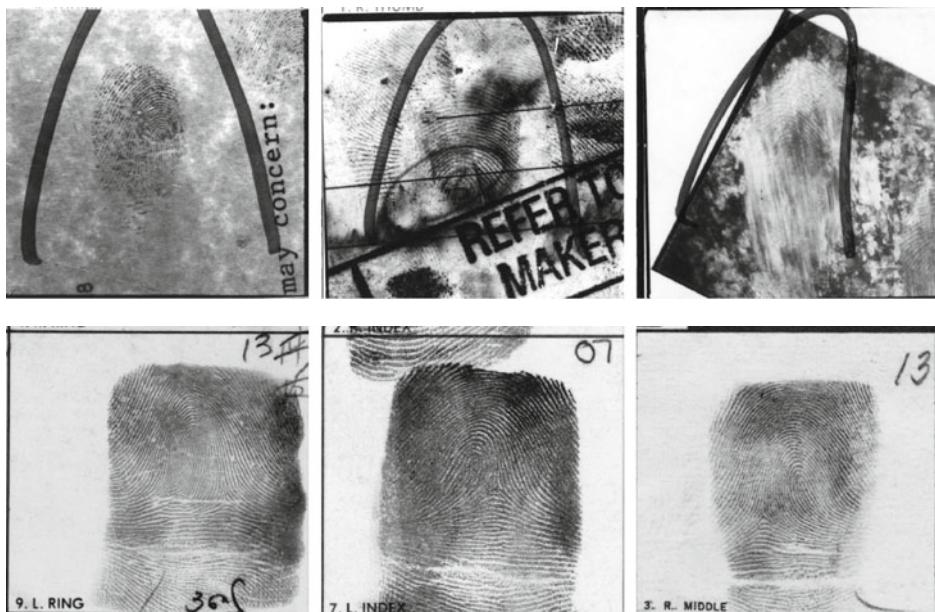


Fig. 6.1 Three latent fingerprints (first row) and mated rolled fingerprints (second row) in NIST DB27. The quality of these latent fingerprints is labeled (from left to right) with “good”, “bad”, and “ugly” by latent examiners

subjects with mates and 20,000 subjects with no mates in the database) against a gallery of 100 K subjects. For latent fingerprint recognition technology, the most recent evaluation is the NIST ELFT-EFS where the best performing automated latent recognition system maxed out at a rank-1 identification rate of 67.2% in searching 1,114 latent fingerprints against a background containing 100,000 exemplar fingerprints (Indovina et al., 2012). Although different performance measures were adopted in these two evaluations, the huge accuracy gap between latent and plain/rolled fingerprint recognition is evident. Therefore, in forensic agencies, in order to improve the identification rate of latent fingerprints, manual annotation is performed by latent examiners to mark the region of interest and minutiae points. Further, a latent examiner needs to compare a latent with the candidate list (typically, 20–50 rolled/plain fingerprints from the database) returned by AFIS to determine the true mate. This is very different from AFIS for ten-print identification, which effectively works in a “lights-out” mode, namely, no fingerprint experts are required in the identification process.

With the gradual improvement in the performance of AFIS, law enforcement agencies have realized that it is possible to utilize automated processing of latent fingerprints to some extent. In 2008 and 2009, NIST in cooperation with AFIS technology providers performed a series of tests referred to as the Evaluation of Latent Fingerprint Technology (ELFT). The results from ELFT Phase II demonstrated that some latent fingerprints can benefit from state-of-the-art automatic feature extraction and matching technologies, thereby reducing some of the human workloads during the latent fingerprint search. It is generally acknowledged that the minutiae marking process performed by a latent examiner requires between 5 and 20 min. With automated feature extraction, latent examiners can use the time saved for more complex tasks. In addition, it allows a faster throughput for latent fingerprint processing, which would otherwise be discarded.

The need for lights-out latent fingerprint recognition has inspired many studies. The research topics include orientation field estimation, segmentation, enhancement, registration, matching, and quality estimation. While the feature extraction and matching steps have been introduced in Chaps. 3 and 4, here we revisit these topics and present approaches specifically designed for latent fingerprints. Note that although these approaches are mainly designed for latent fingerprints, many of them are also applicable to plain or rolled fingerprints. Before we introduce algorithms for latent feature extraction and matching, we briefly introduce the methods of manual latent fingerprint identification. This chapter does not introduce latent fingerprint development technology, which is covered in Champod et al. (2017).

6.2 Latent Fingerprint Recognition by Latent Examiners

6.2.1 ACE-V

Latent examiners conduct latent fingerprint identification according to the ACE-V methodology (Fig. 6.2).

- Analysis: Considering the quality, quantity, and rarity of a latent, the examiner assigns one of the following three values to the latent: Value for Individualization (VID), Value for Exclusion Only (VEO), or No Value (NV). Latent fingerprints of VID and VEO will undergo further analysis, in which features of three levels are manually marked.
- Comparison: The latent is compared with one or more exemplar fingerprints which are collected from persons of interest, such as suspects, or obtained by searching the latent against a large database with an AFIS. Fingerprint features at all three levels (Level 1, Level 2, and Level 3) are compared at this stage.
- Evaluation: Based on evidence collected during comparison and his/her experience, the examiner makes a decision of individualization, exclusion, or inconclusive.
- Verification: Another examiner repeats the ACE process independently to detect possible errors.

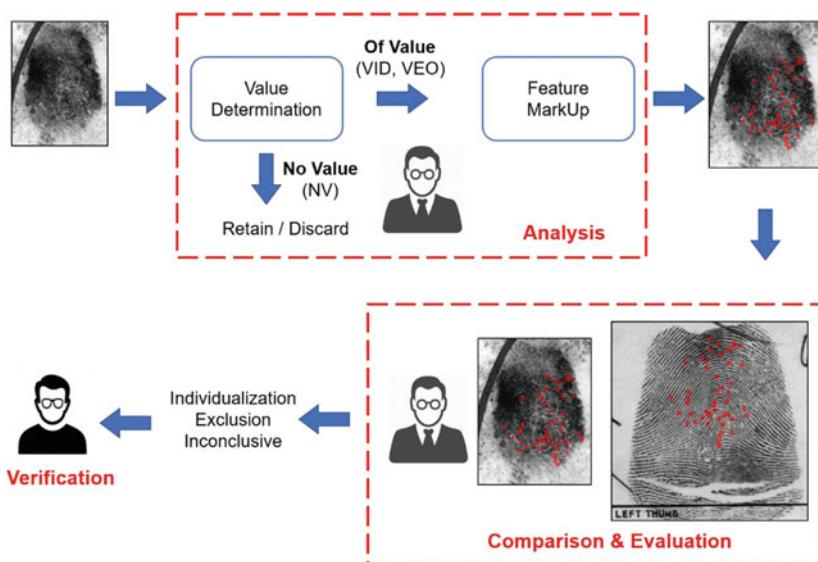


Fig. 6.2 ACE-V methodology. © IEEE. Reprinted, with permission, from Chugh et al. (2018)

Note that ACE-V is a general methodology and the details vary between different publications (Champod et al., 2017). There is not a strict and standardized ACE-V protocol that is followed in every forensic agency. This situation is similar to many different minutiae-based automated fingerprint identification systems, which follow a similar overall methodology but differ significantly in implementation details and performance.

6.2.2 Criticisms

Although it is generally believed that experienced latent examiners are far more accurate than an AFIS in identifying latent fingerprints, latent examiners do make mistakes (Jain & Feng, 2011). There are two types of errors a latent examiner can make: erroneous exclusion (false negative) and erroneous individualization (false positive). An erroneous exclusion occurs when the mated fingerprint of the latent print is in the candidate list reviewed by the latent examiner, but the examiner fails to identify it. An erroneous individualization occurs when a latent print is incorrectly matched to the fingerprint of another subject by the latent examiner. The consequence of erroneous exclusions is that criminals may not be apprehended. On the other hand, the consequence of erroneous individualizations is that wrongful convictions of innocent people may occur. Erroneous individualizations are generally deemed as serious mistakes, while erroneous exclusions are usually seen as less critical. One of the most high-profile cases in which an erroneous individualization was made involves Brandon Mayfield, who was wrongly apprehended in the 2004 Madrid train bombing incident after a latent fingerprint obtained from the bombing site was incorrectly matched with his fingerprint (Fig. 6.3) in the FBI's IAFIS database (OIG, 2006). More cases of erroneous individualization have been brought to

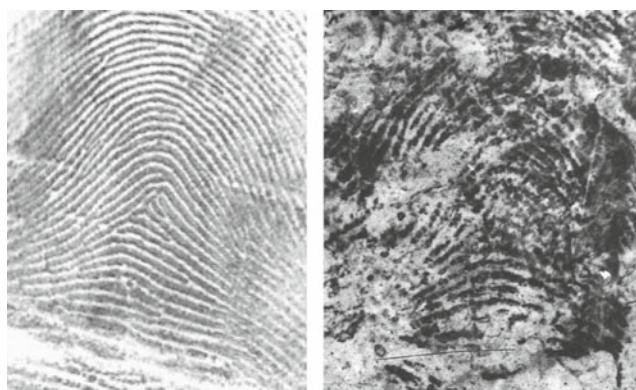


Fig. 6.3 The fingerprint of Mr. Mayfield (left) was incorrectly matched with the latent fingerprint (right) from the crime scene of the Madrid train bombing. © Elsevier. Reprinted, with permission, from Dror et al. (2006)

light by the Innocence Project (Cole, 2005). In a cognitive experiment, Dror et al. (2006) found that latent examiners cannot repeat their individualization decision when they are given a false context, i.e., false hint that they are not mated fingerprints. These incidents and findings have undermined the importance of latent fingerprints as forensic evidence.

In response to the Mayfield case, the FBI tasked a committee to evaluate the fundamental basis for the science of friction ridge pattern analysis (Budowle et al., 2006). The committee concluded that although friction ridge pattern analysis is fundamentally sound, additional studies on friction ridge patterns could improve confidence in the identification results, provide guidelines for more consistent practices in the latent fingerprint community, and provide metrics for evaluating the performance of latent examiners. They suggested several research directions with high priority, including developing statistical models for identification, conducting black box tests of examiners' performance, developing quality metrics for latent fingerprints, and investigating the exculpatory power of latent fingerprints.

In 2009, the US National Research Council (NRC) issued a report (NRC, 2009) that pointed out major problems with the current forensic science discipline, including friction ridge analysis. The main conclusions pertaining to latent fingerprints are summarized as follows:

- The ACE-V methodology does not specify particular measurements or a standard test protocol, and examiners must make subjective assessments throughout.
- The threshold for making an individualization is deliberately kept subjective, so that the examiner can take into account both the quantity and quality of all details in the latent print.
- The outcome of a friction ridge analysis is not necessarily reproducible from examiner to examiner. Furthermore, examiners do not always repeat their own past decisions when the examination is presented in a different context.

6.2.3 Recent Advances

Since the publication of the NRC report, the aforementioned issues have been studied including bias of latent examiners, accuracy and reliability of latent examiners, quality measures of latent fingerprints, and statistical modeling of fingerprints (Champod, 2015). The FBI and Noblis have published a series of papers on these topics, which are summarized in Hicklin (2017). Statistical modeling of fingerprints is presented in Chap. 8.

Ulery et al. (2011) performed a study of the accuracy and reproducibility of decisions by latent examiners. Reproducibility refers to the inter-examiner agreement (whether two examiners reach the same decision on the same latent fingerprint). A total of 164 latent examiners participated in the study and each of them compared 100 pairs of latent

and exemplar prints. Non-mated pairs in this test are much more difficult than randomly selected non-mated pairs since the exemplar prints are obtained by searching the latent against more than 58 million subjects using an AFIS. Since the false positive rate of examiners is very low on randomly selected non-mated pairs, a reliable estimation of false positives on random non-mated pairs would require an impractical number of pairs (say, billions). The major findings of this study are summarized below:

- The study reported a false positive rate of 0.1% and a false negative rate of 7.5%. Given the potentially severe consequences of a false positive, examiners tend to make more false negative decisions.
- Independent examination by a second examiner can detect all false positive errors and most of the false negative errors.
- Most of the false positive errors are latent fingerprints with complex background.
- Among the 356 latent fingerprints which were reviewed by the examiners (each latent was reviewed by 23 examiners, on average), unanimous decisions (either VID or not-VID) were made for only 43% of the latent fingerprints.

In a follow-up study, Ulery et al. (2012) evaluated the repeatability of an examiner's decision (intra-examiner variability) after an interval of 7 months. The major findings of this study are summarized as follows:

- 89.1% of individualization decisions and 90.1% of exclusion decisions were repeated.
- No false positive errors were repeated and 30% of false negative errors were repeated.
- Most of the changes become inconclusive decisions (84.6% of the cases).

Ulery et al. (2016) also studied the inter-examiner variation of minutia markup on latent fingerprints. They found large variations in minutia markups by different examiners, as shown in Fig. 6.4. The primary factor for variations was image clarity. In clear areas of latent fingerprints (low noise), the reproducibility was 82% while in unclear areas (high noise) it was only 46%.

Some researchers have recommended that the ACE process should be performed in a linear manner in order to reduce confirmation bias, which means that people tend to seek and create new evidence in order to verify their preexisting beliefs (Kassin et al., 2013). Confirmation bias is believed to be one of the causes of misidentification in Mayfield's case (OIG, 2006). For example, some studies showed that the minutia markups change between the analysis and comparison stages in the ACE-V process. In particular, Ulery et al. (2015) analyzed changes in latent examiners' markup between analysis and comparison. They found that (1) in the case of individualization, examiners always added or deleted minutiae; (2) changes in minutiae were less common in the case of inconclusive and exclusion; (3) latent fingerprints classified as VEO during analysis were often individualized when compared to a mated exemplar.

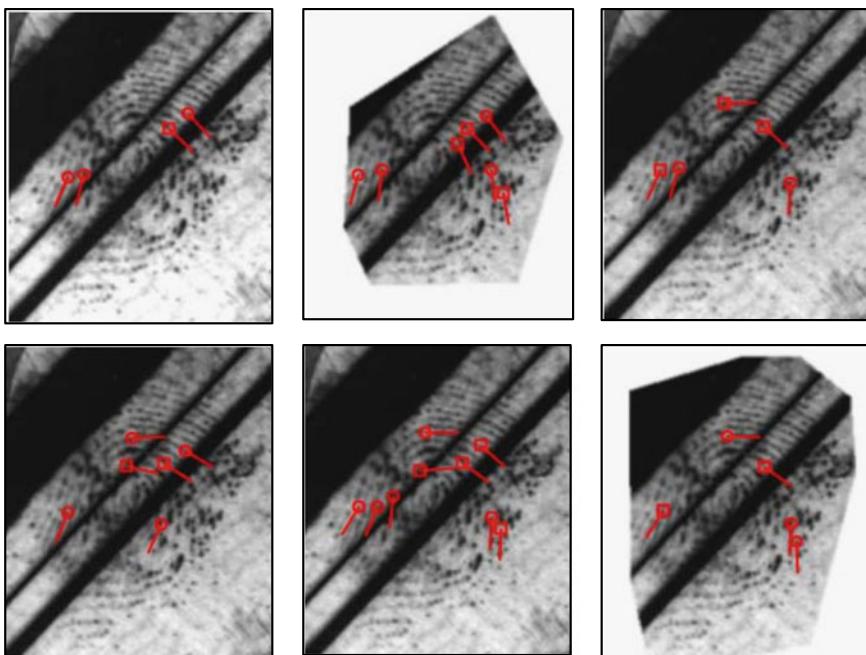


Fig. 6.4 Variability in minutiae markups by six examiners (Arora et al., 2015). © IEEE. Reprinted, with permission, from Arora et al. (2015)

In another two studies (Ulery et al., 2013, 2017), the authors conducted experiments to understand how examiners assess the value of a latent fingerprint, and to analyze the factors associated with exclusion decisions.

6.3 Automated Latent Fingerprint Recognition

Due to the complexity of latent fingerprint recognition and the serious consequences of errors, the latent fingerprint recognition system has both manual and automated components, i.e., it is a hybrid (man in the loop) system. Figure 6.5 shows a typical latent fingerprint identification process. Among the various modules, the collection, feature extraction, and final comparison, all need manual intervention. This is quite different from fingerprint recognition systems used in other applications, such as access control, mobile unlock and payments, and border crossing, where manual intervention is neither required nor affordable due to the real-time nature of the application. As technology evolves, the relative roles of the human and computer change.

The evolution of latent fingerprint identification technology can be roughly divided into three stages:

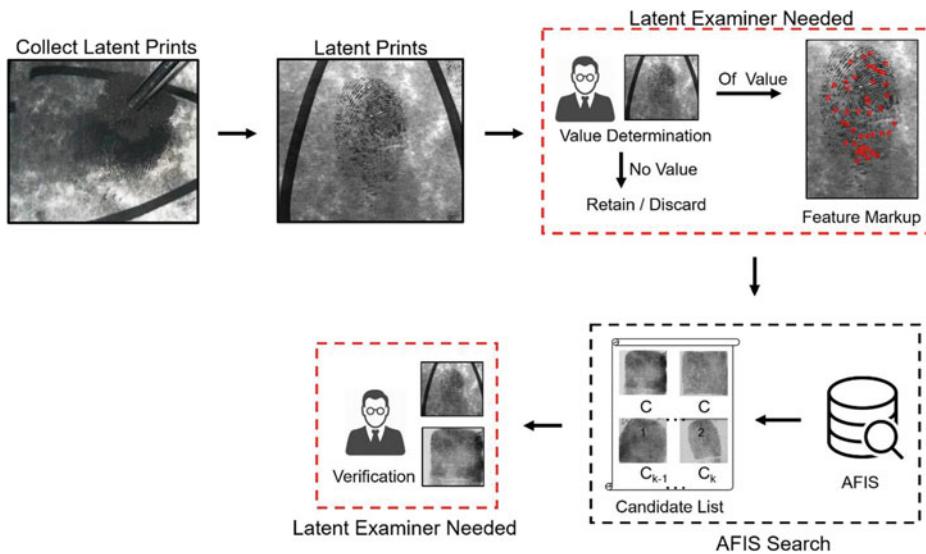


Fig. 6.5 Common process of latent fingerprint identification. In recent years, with the improvement of feature extraction and matching technologies, the feature extraction step can be completely automated for high-quality latent fingerprints

1. *Manual feature extraction.* The early latent fingerprint identification system performed automated latent comparison using manually marked features in latent fingerprints. These systems essentially solved the pain point of manual latent matching, namely to search for a latent's mate in a large database of millions of rolled/plain fingerprints. The matcher is primarily minutiae-based, although additional features (such as orientation field, ridge count) are used as auxiliary features. The original latent fingerprint image itself is not used by the matcher. The main problem of the system at this stage is that manual minutiae extraction is very time-consuming and manually marked minutiae are not very consistent with those extracted by AFIS.
2. *Coexistence of manual and automatic feature extraction.* With the improvements in the performance of feature extraction algorithm on latent fingerprints and the ability of the matcher to utilize both the latent image as well as the latent features, latent matchers can now successfully find true mates for most of the high-quality latent fingerprints. For example, a top-performing COTS latent matcher can achieve 92% rank-1 recognition rate on the NIST DB27 “good” quality latent fingerprint subset with a background of 100 K rolled fingerprints (Cao et al., 2020). However, manual feature markup is still necessary for two reasons. First, for low-quality latent fingerprints, the algorithm’s performance is not satisfactory (the same COTS latent matcher mentioned above has only 41% rank-1 recognition rate for the “ugly” latent fingerprint subset of NIST DB27). Second, a fusion of manually marked features and automatically extracted features

has higher performance as shown in the ELFT-EFS evaluation. In order to reduce the workload on latent examiners without sacrificing the recognition accuracy, it is particularly important to automatically determine whether a latent fingerprint needs manual intervention.

3. *Fully automated latent recognition.* With the continuous improvement in latent AFIS performance, a majority of latent fingerprints may not need manual intervention. The latent matcher can not only generate a short list of candidates, but also output the confidence associated with candidate fingerprints, which is convenient for latent examiners to make decisions. Only in some special cases, will the human intervention be required. For example, if there are multiple overlapping fingerprints among which the fingerprint of interest is unknown; or if there is unusual image distortion that requires knowledge from the crime scene and cannot be obtained from the fingerprint image. At this stage, the focus of latent examiners will move forward to collect as many latent prints as possible at the scene and improve the quality of latents.

In the first two stages above, the latent fingerprint identification system is a human-machine hybrid system. This is similar to artificial intelligence technologies in critical applications such as medical diagnosis and autonomous driving. The human-machine hybrid phase is likely to last for a long time due to the complexity of the task and the consequence of wrong decisions. However, considering the rapid improvement in latent matching algorithm performance, we are still optimistic that a fully automated latent fingerprint identification phase with a high recognition rate and reliable confidence measure will become available.

The following two sections introduce the methods for latent fingerprint feature extraction and matching that have emerged in the past ten years and that are available in the public domain.

6.4 Feature Extraction

6.4.1 Challenges

There are several obvious differences between latent fingerprints and live-scan or inked fingerprints, which make it difficult to accurately extract features from latent fingerprints. For example, (1) high levels of noise in latent fingerprints make it difficult to extract clear ridges; (2) small latent friction ridge area and poor contrast between foreground and background make it difficult to extract a precise segmentation; (3) the large nonlinear distortion due to pressure variations between the finger and object surface changes the consistency of local orientation field.

Figure 6.6 shows the results of feature extraction with [VeriFinger SDK 12.0](#) for a latent fingerprint and an inked rolled fingerprint in NIST DB27, and a plain fingerprint

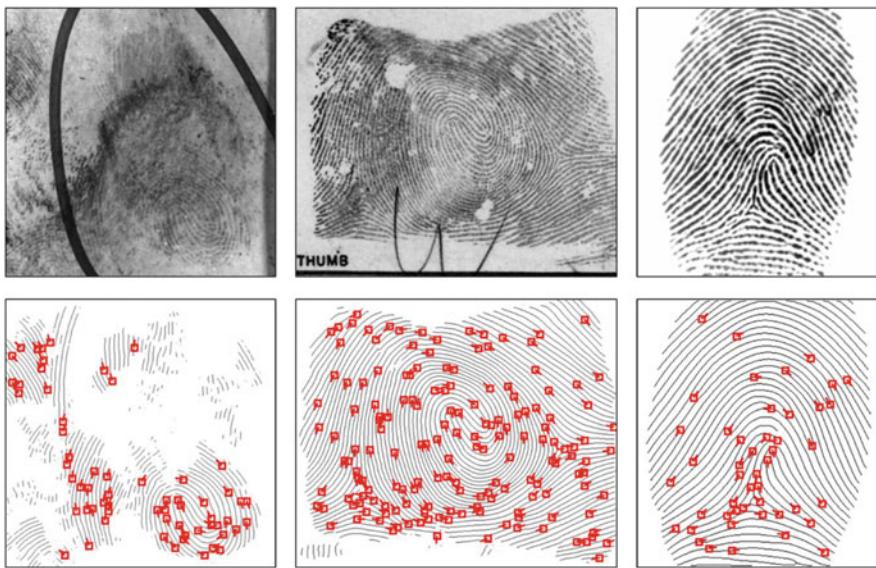


Fig. 6.6 Ridge and minutiae extraction from (left to right) a latent fingerprint and an inked rolled fingerprint in NIST DB27 and a plain fingerprint in FVC2002 DB1 by a COTS fingerprint algorithm (VeriFinger SDK 12.0)

in FVC2002 DB1. It can be seen that due to a large amount of background noise, it is very difficult to accurately extract features in latent fingerprint images. A large number of false minutiae are extracted and some real minutiae are missed. Obviously, the recognition performance based on such inaccurate features is also poor.

The following subsections present advances in latent fingerprint feature extraction in order of coarse to fine feature levels.

6.4.2 Pose Estimation

As discussed in Sect. 3.5.5, the pose of a fingerprint is defined by the center point (x, y) and the direction θ (Fig. 6.7). The center of a fingerprint can be regarded as the point corresponding to the geometric center on the front of the fingertip (e.g., a core or focal point), while the direction of the fingerprint is regarded as the direction the finger is pointing at.

The obvious application of fingerprint pose is for fingerprint matching and indexing, as discussed in Sect. 3.5.5. If pose can be consistently estimated from latent and corresponding rolled fingerprints, it is beneficial for both the matching accuracy and speed. Another

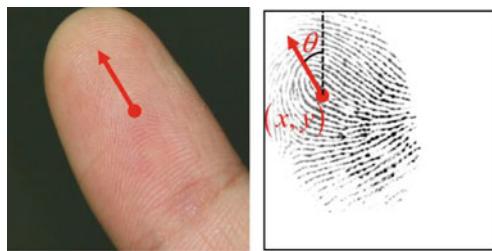


Fig. 6.7 The pose of a finger is shown on a finger photo and a fingerprint image. © IEEE. Reprinted, with permission, from Yang et al. (2014)



Fig. 6.8 The center is often missing, occluded or distorted in latent fingerprints

application of fingerprint pose is that location-related prior information of fingerprints can be used in extracting the orientation field.

In latent fingerprints, it is difficult to estimate the accurate fingerprint pose due to a large amount of distortion and noise. As shown in Fig. 6.8, the foreground area of the fingerprint is limited, and the central area is missing or seriously occluded, making it difficult to estimate an accurate fingerprint pose.

To deal with the problem of pose estimation of latent fingerprints, Yang et al. (2014) studied the distribution of the ridge orientation in different regions of fingerprints and established a statistical model based on the local features. The whole system consists of two stages: in the offline stage, they learn the spatial distribution of real fingerprint orientation blocks in different regions related to the fingerprint pose and then establish a statistical model of prototype orientation blocks. In the online stage, each block of an input fingerprint makes a prediction of the fingerprint center. These predictions are then summed, and the peak value is selected as the final pose of the input fingerprint.

In the offline stage (Fig. 6.9), the authors first selected 398 high-quality fingerprints from the NIST DB4 rolled fingerprint database as the training set, and then manually marked their poses and orientation fields. For each training orientation field registered with respect to its pose, a 4×4 window slides across the image from top to bottom

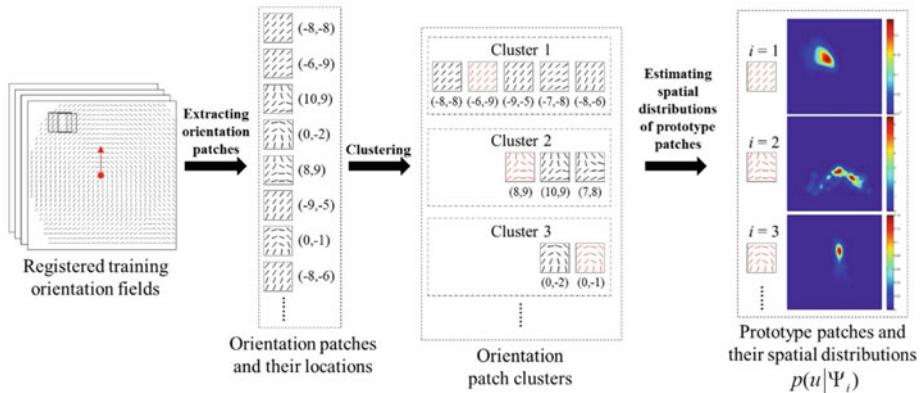


Fig. 6.9 In the offline stage, the prototype orientation patches and their spatial distributions are learnt from a set of registered training orientation fields (Yang et al., 2014). © IEEE. Reprinted, with permission, from Yang et al. (2014)

and left to right. If all the orientation elements in the sliding window are valid, then the orientation patch and its center's coordinates are added to the real orientation block training set. All the orientation patches were clustered into 200 classes by the K-center clustering method, and the 200 cluster centers form the prototype orientation patches. Finally, the spatial distribution of each prototype orientation patch is calculated.

In the online stage (Fig. 6.10), for an input fingerprint, first the initial orientation field is calculated by short-time Fourier transform (STFT). Then a 4×4 sliding window slides across the orientation field from top to bottom and left to right. In each location, the

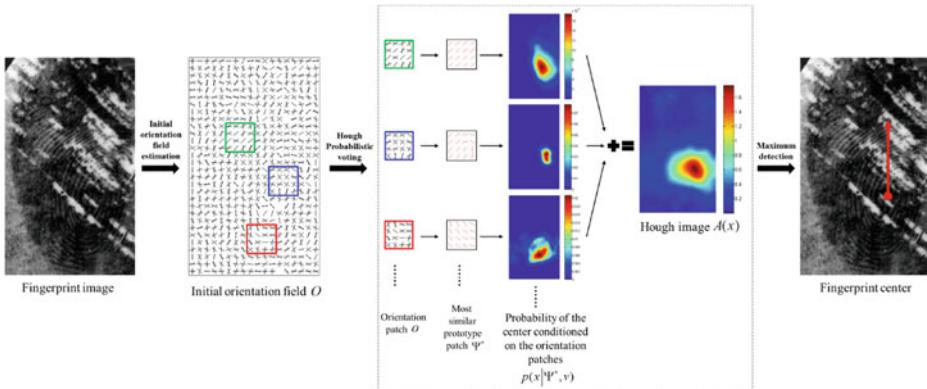


Fig. 6.10 Flowchart of estimating the finger center of an upright latent fingerprint (Yang et al., 2014). © IEEE. Reprinted, with permission, from Yang et al. (2014)

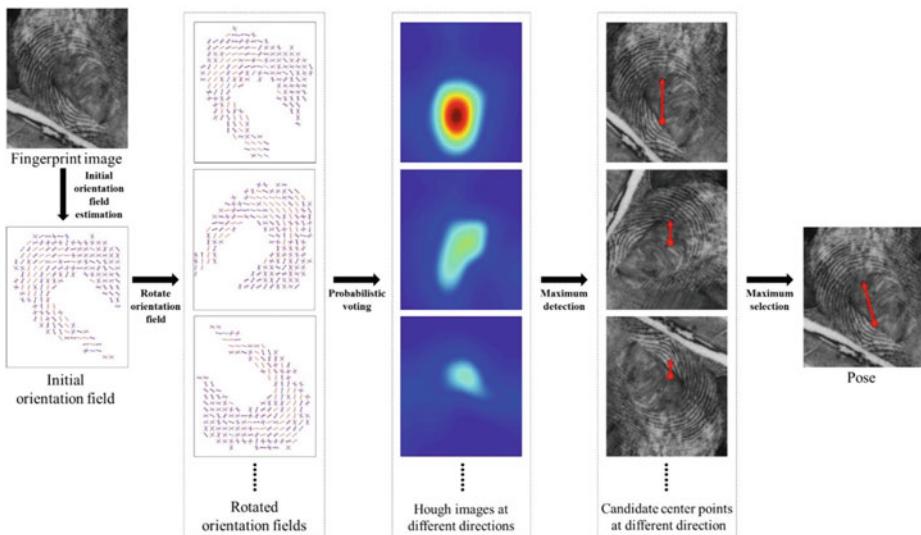


Fig. 6.11 Pose estimation of a latent fingerprint with unknown direction (Yang et al., 2014). The initial orientation field is rotated by a set of possible angles; the finger center estimation procedure is then applied to detect the center at each angle; and finally, the one with the largest confidence value is chosen. © IEEE. Reprinted, with permission, from Yang et al. (2014)

most similar prototype orientation patch is found and votes for the fingerprint center are obtained by the spatial distribution of this prototype. By adding up the votes from all locations of the input fingerprint, the Hough image can be calculated, and the peak value is selected as the final estimation of the fingerprint center. Finally, the orientation field is rotated at different angles and the steps above are repeated for each angle to estimate centers and the corresponding confidence values (see Fig. 6.11). The final pose is the center and the angle with the highest values.

Some results of pose estimation are shown in Fig. 6.12. As can be seen, some of the latent fingerprints are missing the central region. Hough voting algorithm can deal with this situation quite well. It is worth mentioning that, pose estimation of latent fingerprints is still an open problem. Much improvement is needed if it is to be used for latent fingerprint matching or indexing.

6.4.3 Foreground Segmentation

Latent fingerprint segmentation means estimating the friction ridge region of a latent fingerprint image. Accurate segmentation is crucial for latent fingerprint feature extraction. In a latent fingerprint image, the effective Region of Interest (RoI) is often limited in the

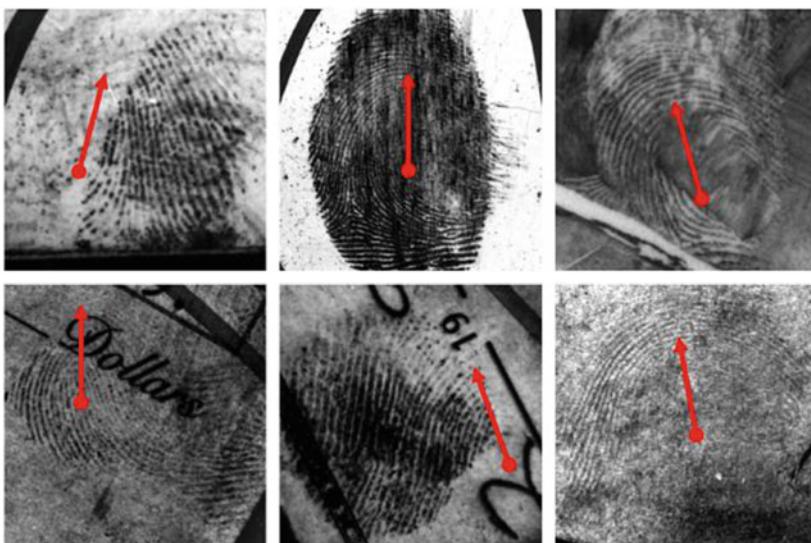


Fig. 6.12 The results of pose estimation for some of the latent fingerprints in NIST DB27 (Yang et al., 2014). © IEEE. Reprinted, with permission, from Yang et al. (2014)

area; most of the latent image consists of a background area containing a lot of noise. If the entire latent image is input into the automatic fingerprint identification system, a large number of false minutiae will be generated, resulting in severe degradation of identification performance.

However, due to the extremely low signal-to-noise ratio of latent fingerprints, the ridge structure in a latent fingerprint is often incomplete. Furthermore, there are often structures highly similar to ridge patterns in the background, making it difficult for traditional algorithms to effectively separate the friction ridge region from the background region. Total-variation (Sect. 3.2.3) and deep learning-based methods (Sect. 3.2.4) have been shown to be well suited for latent fingerprints. Here we describe a deep learning-based foreground segmentation method in detail (Nguyen et al., 2018). Note that some latent feature extraction algorithms can output a segmentation mask while extracting the orientation field or minutiae. These methods will be described in later sections.

Nguyen et al. (2018) presented a simple but effective method for latent fingerprint segmentation, called SegFinNet. SegFinNet takes a latent image as an input and outputs a binary mask highlighting the ridge pattern. The algorithm combines fully convolutional neural networks and detection-based approaches to process the entire input latent image in one shot instead of processing latent patches with a sliding window.

SegFinNet uses Faster RCNN as its backbone while its head layer comprises atrous transposed convolution layers (as shown in Fig. 6.13). The authors utilize a combination of a non-warp region of interest technique, a fingerprint attention mechanism, and

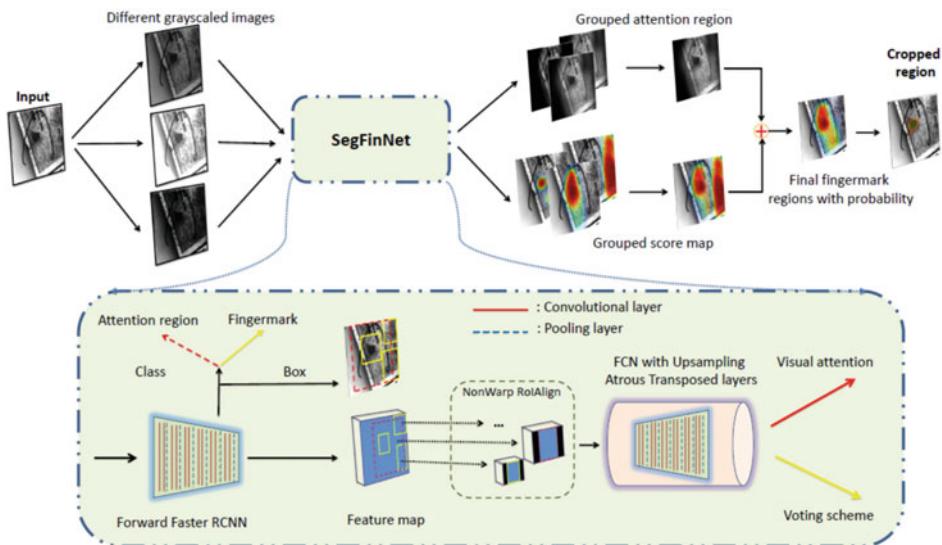


Fig. 6.13 Network architecture of SegFinNet (Nguyen et al., 2018). © IEEE. Reprinted, with permission, from Nguyen et al. (2018)

a fusion voting and a feedback scheme to take advantage of both the deep information from neural networks and the shallow appearance of fingerprint domain knowledge. The NonWarp-RoIAlign is proposed to obtain precise segmentation while mapping the region of interest (cropped region) in the feature map to the original image. The visual attention technique is designed to focus only on friction ridge regions in the input image, which addresses the problem of “where to look” (attention). The feedback scheme with weighted loss is utilized to emphasize the difference in importance of different objective functions (foreground–background, bounding box, etc.), and a majority voting fusion mask is proposed to increase the stability of the cropped mask while dealing with different qualities of latent fingerprint images.

The experiments were carried out on NIST DB27 and WVU latent fingerprints databases. To evaluate the cropping accuracy, the authors introduced pixel-wise MDR (missing detection rate) and FDR (false detection rate) metrics. SegFinNet provides lower MDR and FDR than previously published methods. The MDR is 2.57% and 12.15%, while the FDR is 16.36% and 5.30% on NIST DB27 and WVU databases, respectively. Some examples of segmentation are shown in Fig. 6.14. A good segmentation also improves the matching performance as shown by the authors.

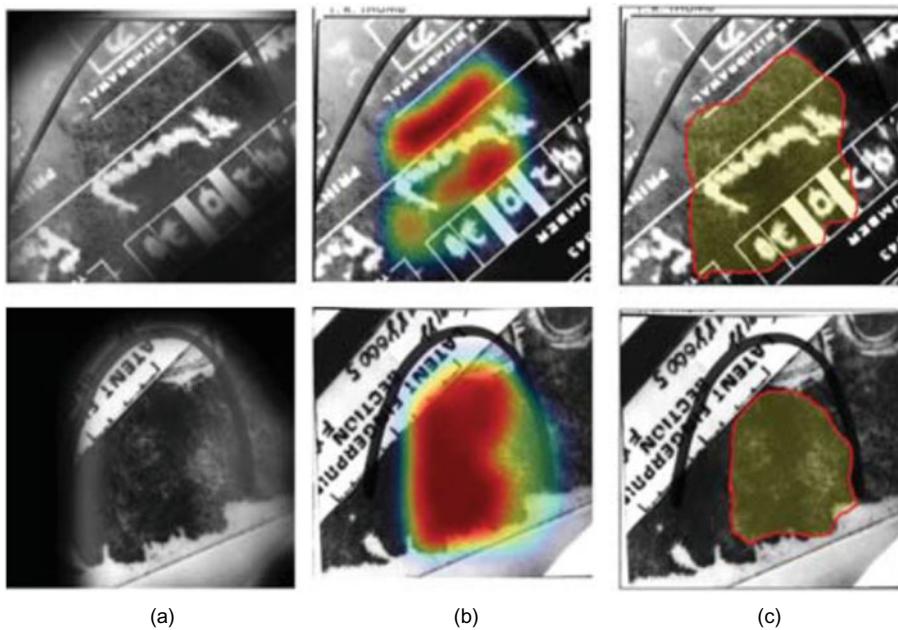


Fig. 6.14 SegFinNet (Nguyen et al., 2018) with visual attention mechanism for two different input latent fingerprints, one per row: **a** Focused region from visual attention module; **b** Original latent fingerprints overlaid with a heat map showing the probability of occurrence of friction ridges (from high to low); **c** Binary mask (boundary marked in red). © IEEE. Reprinted, with permission, from Nguyen et al. (2018)

6.4.4 Local Ridge Orientation Estimation

As stressed in Chap. 3, the orientation field extraction plays an important role in the subsequent processing of fingerprints. For example, in Gabor filters-based fingerprint enhancement, local ridge orientation is one of the two key parameters of the filters. With an accurate orientation field, Gabor filtering can effectively enhance the ridge structure. On the contrary, with incorrect orientation field estimation, the true ridge structure can be incorrectly altered by the filters, resulting in inaccurate features and hence serious degradation of matching performance.

Traditional orientation field extraction algorithms estimate an initial orientation field using local information and then smooth it to reduce the effect of noise. These orientation field estimations are heavily affected by the noise in latent fingerprints. For noisy latent fingerprints, these approaches cannot restore the correct orientation field. Learning-based methods have been shown to be superior thanks to the prior information they learn in an early offline stage. Dictionary-based approaches became popular as an effective method to regularize very noisy local ridge orientations. CNN-based techniques have been shown

to be even more accurate for this task. A brief review of learning-based local orientation estimation is reported in Sect. 3.3.6. Hereafter, we focus on the local dictionary method (Yang et al., 2014) and the CNN-based method (Cao & Jain, 2015).

Yang et al. (2014) improved the dictionary-based method in Feng et al. (2013). They noticed that ridge orientations in different regions of fingerprints have different characteristics as shown in Fig. 6.15. Considering the prior knowledge of the orientation fields in different regions in a fingerprint, they established different dictionaries through a statistical model for each area. For an input fingerprint, its pose is estimated using the approach

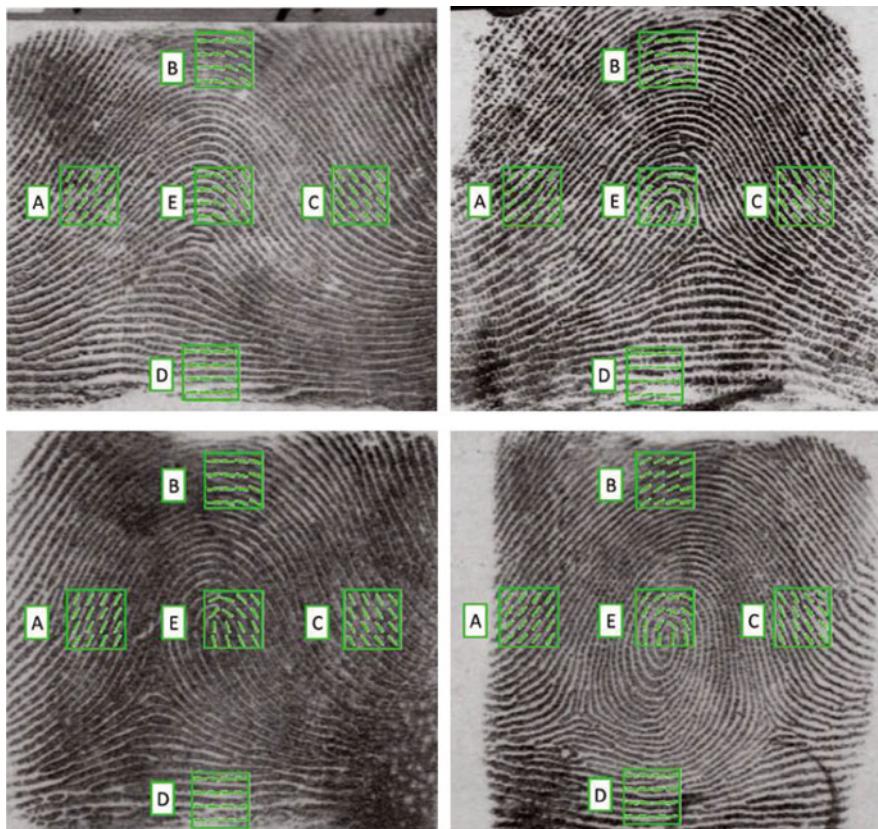


Fig. 6.15 Ridge orientations in different regions of fingerprints have different characteristics as shown in these four fingerprints of different pattern types. Ridge orientations in the central region (patch E) of fingerprints are very diverse, while the ones in the peripheral region (patches A, B, C, and D) lack variety and are independent of fingerprint pattern types. In addition, the orientation patches in the four different peripheral regions are different from each other (Yang et al., 2014). © IEEE. Reprinted, with permission, from Yang et al. (2014)

described in Sect. 6.4.2. The fingerprint is then registered to the uniform coordinate system. Then each initial orientation block in the foreground area is used for retrieving similar blocks in the corresponding dictionary at a specific location to get the query candidate set. Finally, the authors use an optimization framework to get the final orientation field.

Specifically, in the offline stage (see Fig. 6.16), local orientation block dictionaries are constructed at different positions in the fingerprint coordinate system. The steps are as follows:

- The pose estimation algorithm (in Sect. 6.4.2) is used to register all the training orientation fields to the fingerprint coordinate system. In the fingerprint coordinate system, the union set of all the foreground of training orientation field is defined as the effective area of the localized dictionaries \mathcal{F} .
- For any position u in \mathcal{F} , all the valid orientation blocks are collected to constitute the training set \mathcal{T}_u , and the first orientation block in \mathcal{T}_u is selected into the local dictionary \mathcal{D}_u .
- Take out a training orientation block that has not been checked from \mathcal{T}_u , and check whether it is similar enough to any existing orientation block in \mathcal{D}_u . If not, it is selected and inserted into \mathcal{D}_u , otherwise, it is discarded.
- Repeat the previous steps until all orientation blocks in \mathcal{T}_u have been checked.
- Expand each orientation block in \mathcal{D}_u by increasing or decreasing some small degrees of all orientation elements in each orientation block at the same time to get new blocks and add these new blocks into \mathcal{D}_u .

In the online stage (see Fig. 6.16), the fingerprint image is first divided into non-overlapping image blocks of 8×8 pixels, and the corresponding orientation of each image block is estimated based on the short-time Fourier transform (STFT) method. The initial orientation field \mathbf{O} is estimated, then it is registered to the fingerprint coordinate system with respect to its pose and denoted as \mathbf{O}_f . For each position (e.g., u) of the registered initial orientation field \mathbf{O}_f , its n_c most similar candidate blocks are retrieved from their corresponding dictionary \mathcal{D}_u . An energy function is defined by (1) the similarity between all the selected candidate blocks and the initial orientation field as well as (2) the compatibility between the candidate blocks,

$$E(\mathbf{r}) = E_s(\mathbf{r}) + \omega_c E_c(\mathbf{r}),$$

where \mathbf{r} denotes the indices of determined candidate patches, $E_s(\mathbf{r})$ denotes the similarity term, $E_c(\mathbf{r})$ denotes the compatibility term, and ω_c is the weight. The energy function is minimized using the loopy belief propagation algorithm (Blake et al., 2011). The final orientation field estimation result is obtained by combining the candidate patches which minimize the energy $E(\mathbf{r})$.

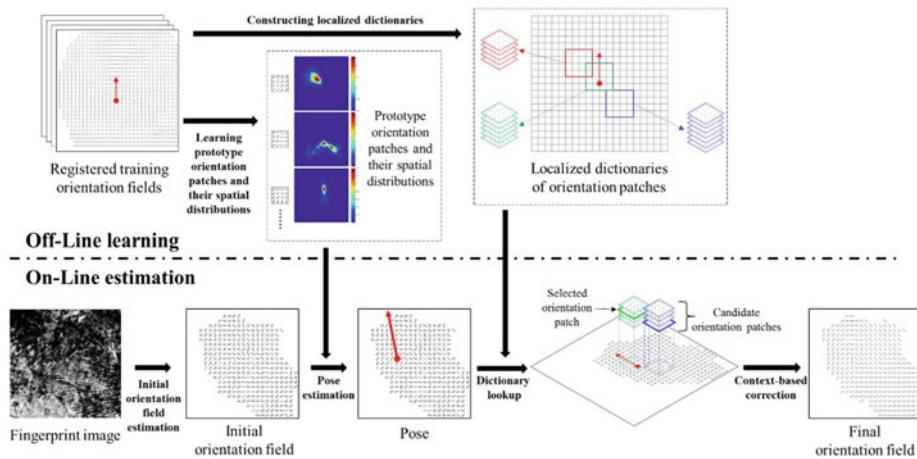


Fig. 6.16 The flowchart of orientation field estimation using localized dictionaries (Yang et al., 2014). © IEEE. Reprinted, with permission, from Yang et al. (2014)

Figure 6.17 demonstrates that this algorithm is robust to noise in latent fingerprints. A potential risk with this algorithm is that it applies pose estimation before orientation field estimation and thus errors in pose estimation may affect the latter steps. However, the experimental results of orientation field estimation show that the influence of inexact pose on orientation field estimation is limited. Figure 6.18 shows an example where the

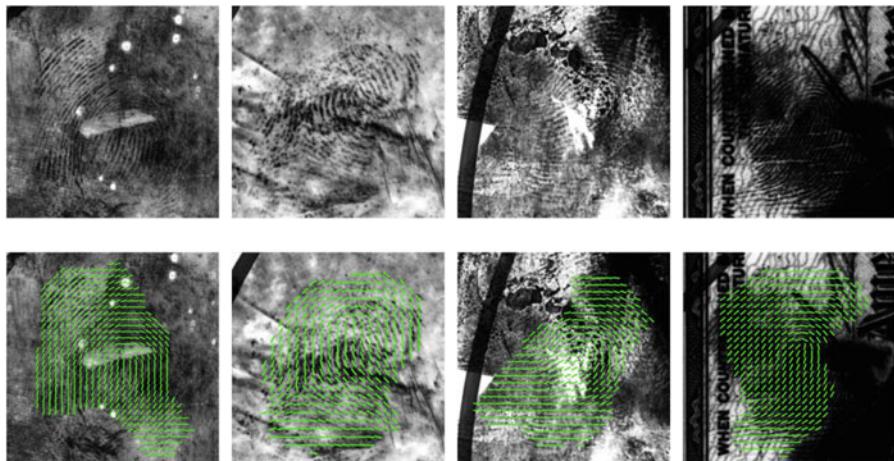
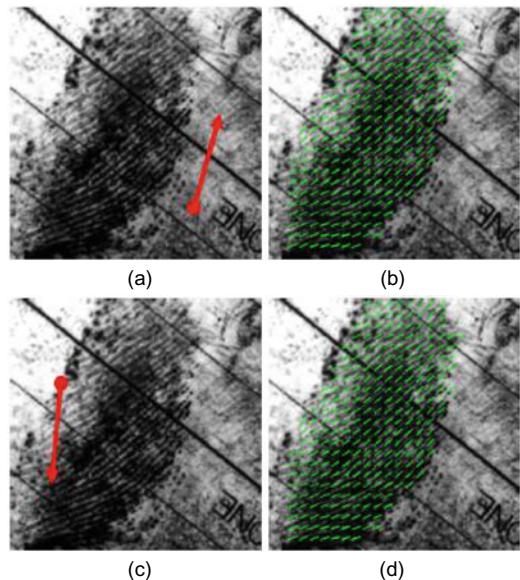


Fig. 6.17 The results of localized dictionary-based orientation field estimation method (Yang et al., 2014). © IEEE. Reprinted, with permission, from Yang et al. (2014)

Fig. 6.18 An incorrect estimate of the pose does not necessarily impact the estimation of the orientation field (Yang et al., 2014). **a** True pose. **b** Orientation field is estimated based on the true pose. **c** Incorrectly estimated pose. **d** Orientation field is estimated based on the incorrect pose. © IEEE. Reprinted, with permission, from Yang et al. (2014)



wrong estimate of the pose does not impact the estimation of the orientation field.

Yang et al. (2015) extended the above method to perform latent fingerprint detection and segmentation, without any manual markup. Multiple potential latent fingerprints are detected using a sequential pose estimation algorithm. Then, the full orientation field and confidence map of each detected fingerprint is estimated based on localized dictionaries lookup. Finally, the boundary of each latent fingerprint is delineated by analyzing its confidence map.

Cao and Jain (2015) posed orientation estimation of a fingerprint patch as a classification problem and proposed a convolutional neural network (ConvNet) based approach for latent orientation field estimation. Given image patches extracted from a latent, their orientation patches are predicted by the trained ConvNet and quilted together to form the orientation field of the whole latent fingerprint.

Specifically, using the NIST fingerprint image software (Watson et al. 2007), all orientation fields with a block size of 16×16 pixels are first obtained from the NIST DB4 database which contains about 400 rolled fingerprints for each of the five fingerprint types, i.e., arch, tented arch, left loop, right loop, and whorl. Orientation patches with the size of 10×10 blocks are selected from these orientation fields. A fast K-means clustering approach is then adopted to cluster these collected orientation patches into 128 clusters, also named orientation patterns. A subset of orientation patterns is shown in Fig. 6.19. From another larger rolled fingerprint database, NIST DB14, a large number of fingerprint patches with the size of 160×160 pixels are selected and assigned to a corresponding cluster by computing the orientation similarity with each orientation pattern. Thus, for

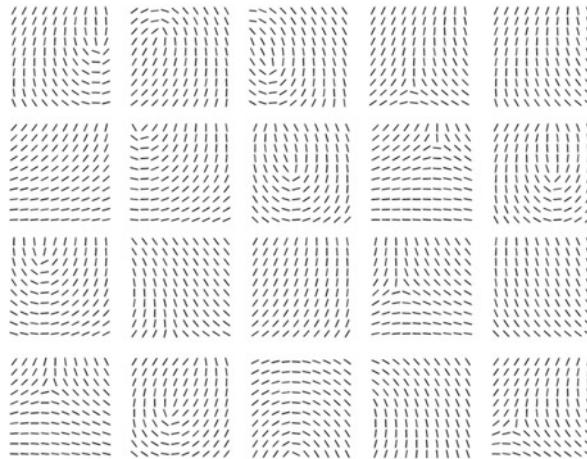


Fig. 6.19 A subset of the 128 orientation patterns learned from the NIST DB4 database (Cao & Jain, 2015). © IEEE. Reprinted, with permission, from Cao and Jain (2015)

each orientation pattern class, 10,000 fingerprint patches are selected for ConvNet training. These fingerprint patches are corrupted by line-like texture noise to simulate latent fingerprints before they are fed into the ConvNet. The outline of the ConvNet is shown in Fig. 6.20.

Given a latent image, the orientation field is estimated as follows (shown in Fig. 6.21): (1) a preprocessing step is used to remove large scale background noise and enhance the potential ridge structure of the remaining texture component in the latent; (2) the preprocessed latent is divided into overlapping patches and each patch is fed to the trained

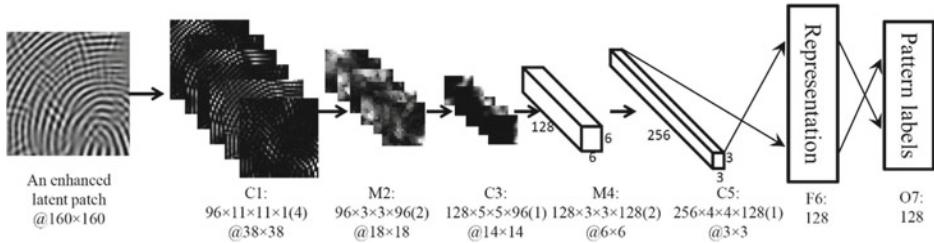


Fig. 6.20 Outline of the ConvNet architecture (Cao & Jain, 2015). There are three convolutional layers (C1, C3, and C5), two max-pooling layers (M2 and M4), a fully-connected layer (F6), and an output layer (O7). The ReLU activation function is used for each layer except the output layer. Dropout regularization is used in F6 to encourage sparsity of the neurons and to avoid overfitting. © IEEE. Reprinted, with permission, from Cao and Jain (2015)

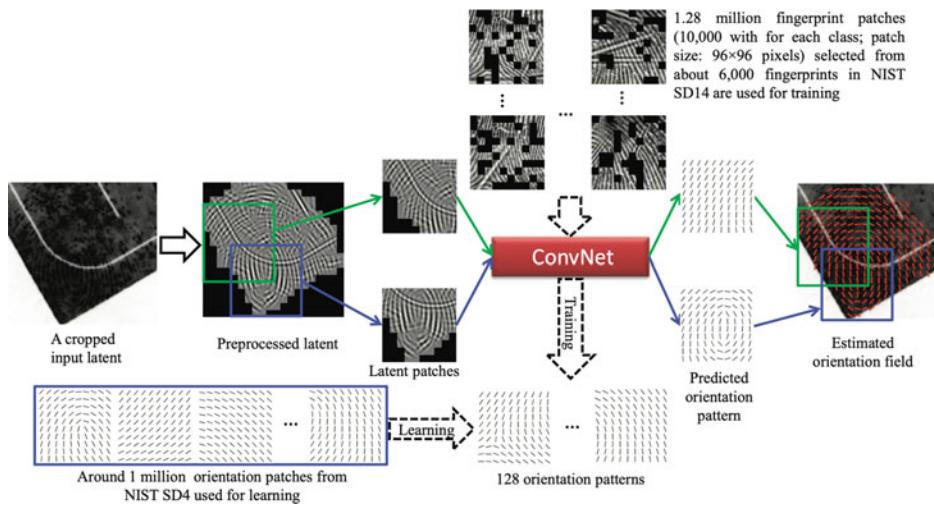


Fig. 6.21 The flowchart of ConvNet based orientation field estimation algorithm (Cao & Jain, 2015). © IEEE. Reprinted, with permission, from Cao and Jain (2015)

ConvNet to predict its orientation pattern; (3) the predicted orientation patterns of all the latent patches are quilted together to form the orientation field of the whole latent.

Most studies on orientation field estimation for latent fingerprints performed two types of evaluation: the accuracy of orientation field estimation and identification performance based on enhanced fingerprints using estimated orientation field. Due to the difference in background databases and fingerprint matchers, it is not easy to compare the identification performances in different studies. Accuracies of orientation field estimation are comparable since most of them used the same manually marked orientation fields in Feng et al. (2013) as ground truth. Table 6.1 compares the performance of different orientation field estimation algorithms in terms of the average Root Mean Square Deviation (RMSD),

Table 6.1 Average Root Mean Square Deviation (RMSD) of different orientation field estimation algorithms on NIST DB27 latent database and its three subsets

Algorithms	All	Good	Bad	Ugly
Global dictionary (Feng et al., 2013)	18.44	14.40	19.18	21.88
Local dictionary (Yang et al., 2014)	14.35	11.15	15.15	16.85
RidgeDict (Cao et al., 2014)	19.53	15.34	20.70	22.68
ConvNet (Cao & Jain, 2015)	13.51	10.76	13.94	16.00
ExhaustiveSearch (Yin et al., 2018)	13.01	10.85	13.99	14.27
FPRefNet (Duan et al., 2021)	12.16	9.87	12.83	13.85

which is adopted in FVC-onGoing FOE (Fingerprint Orientation Extraction) benchmark. Best performance is obtained by recent deep learning-based methods.

6.4.5 Overlapping Fingerprint Separation

The problem of overlapping fingerprints is also common in latent fingerprints. Overlapping fingerprints refer to the existence of two or more fingerprints at the same location, the ridge information of other fingerprints will also cause adverse effects on the identification of the main fingerprint. To deal with this, some researchers proposed automatic overlapping fingerprint separation algorithms, with a focus on obtaining the orientation field of each component fingerprint.

Chen et al. (2011) proposed an orientation field estimation algorithm to separate overlapped fingerprints into component fingerprints. They first estimate an initial orientation field of the given image with overlapped fingerprints, which has two orientation elements in the overlapping region and one orientation element in the non-overlapping region. Then they separate the initial orientation field into component orientation fields using a relaxation labeling technique. The flowchart of this algorithm is given in Fig. 6.22. They also proposed to utilize fingerprint singularity information to further improve the separation performance.

Zhao and Jain (2012) improved the robustness of overlapping fingerprints separation, particularly for low-quality images. Their algorithm reconstructs the orientation fields of

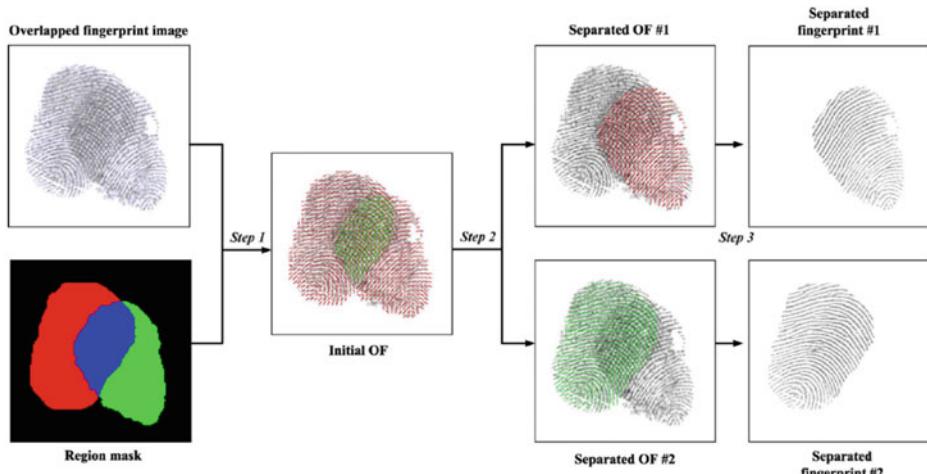


Fig. 6.22 Flowchart of separating overlapped fingerprints (Chen et al., 2011). © IEEE. Reprinted, with permission, from Chen et al. (2011)

component prints by modeling fingerprint orientation fields with the orientation cues of component fingerprints, which are manually marked by fingerprint examiners.

Feng et al. (2012) proposed an orientation field estimation algorithm (called the basic algorithm) for latent overlapping fingerprints whose core is the constrained relaxation labeling algorithm. They further proposed improved versions of the basic algorithm for two special but frequent cases: (1) the mated template fingerprint of one component fingerprint is known and (2) the two component fingerprints are from the same finger. In both cases, further constraints are used to reduce ambiguity in relaxation labeling.

6.4.6 Ridge Enhancement and Minutiae Detection

Ridges and minutiae are closely related features and thus their extraction is discussed together. If orientation extraction is reliable, contextual filtering with Gabor filters (described in Sect. 3.6.2) is effective for enhancing ridges in latent fingerprints too. An average ridge frequency can be used if the ridge pattern is too noisy (instead of a local frequency estimation). If ridges are correctly enhanced then classical binarization, thinning, crossing number techniques (described in Sect. 3.7.1) can be used for minutiae detection.

Recently, deep learning-based techniques were specifically introduced for latent fingerprint enhancement and minutiae extraction as surveyed in Sects 3.6.4 and 3.7.3, respectively. Here, we describe two approaches (Tang et al., 2017; Dabouei et al., 2018) in more detail.

Tang et al. (2017) proposed to design a deep convolutional network by combining domain knowledge of fingerprints and the representation ability of deep learning. In terms of orientation estimation, segmentation, enhancement, and minutiae extraction, traditional (image processing-based) methods that perform well on rolled/plain fingerprints are transformed into convolutional manners and integrated as a unified plain network with fixed weights (shown in Fig. 6.23). The network, named FingerNet, is then expanded to enhance its representation ability and the weights are released to learn complex background variance from data while preserving end-to-end differentiability.

The detailed network architecture is shown in Fig. 6.24. First, to deal with an input fingerprint image, pixel-wise normalization is adopted to fix the mean and variance of the input image. After image normalization, the whole network can be divided into three parts according to different tasks: orientation estimation and segmentation, enhancement, and minutiae extraction.

The backbone of the orientation and segmentation component is a VGG-structure network, which consists of several convolution-BN-pReLU blocks and max-pooling layers. After the basic feature extraction, the atrous spatial pyramid pooling (ASPP) layer is adopted to get multi-scale information. The rates for atrous convolution are 1, 4, and 8. Subsequently, parallel orientation regression is carried out on feature maps of each scale to directly predict the probabilities of 90-discrete angles for each input pixel, getting the

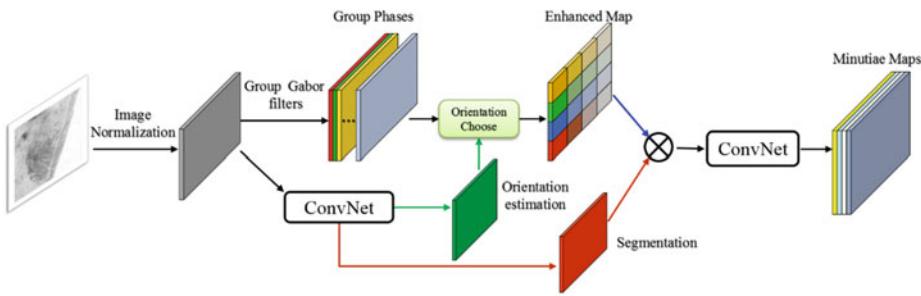


Fig. 6.23 Pipeline of plain FingerNet (Tang et al., 2017). Traditional methods consisting of orientation estimation, segmentation, Gabor enhancement, and extraction are transformed into convolutional manners and integrated as a network. © IEEE. Reprinted, with permission, from Tang et al. (2017)

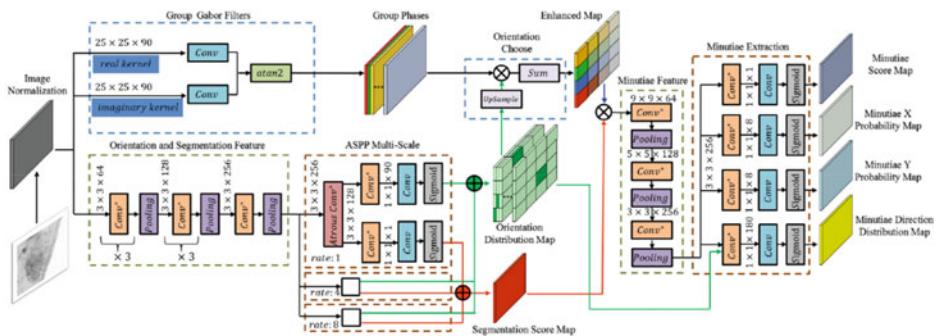


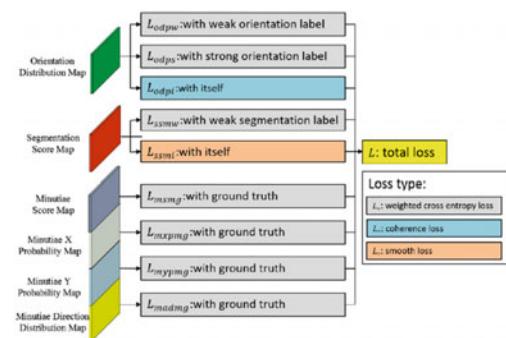
Fig. 6.24 The detailed FingerNet architecture (Tang et al., 2017). Expanded from the plain FingerNet, it can be trained end-to-end. © IEEE. Reprinted, with permission, from Tang et al. (2017)

orientation distribution map. Segmentation regression is carried out to predict the probability of each input pixel to be the region of interest, providing the segmentation score map.

Gabor enhancement is directly adopted as the enhancement part. The ridge frequency is fixed, and ridge orientation is discretized to 90 discrete angles, corresponding to the orientation distribution map. The group phases are multiplied by the upsampled orientation distribution map to get the final enhanced fingerprint image. Specifically, the parameters of Gabor filters are set trainable and are fine-tuned in the training process.

After enhancement, the enhanced fingerprint image is sent to the minutiae extraction component. The backbone of this component is also a VGG-structure network followed by an ASPP layer. After feature extraction, the minutiae extraction part outputs four different maps to fulfill the network requirements. The first map is a minutiae score map, which represents the probability of each 8×8 block to contain a minutia. The second and

Fig. 6.25 The loss functions for FingerNet (Tang et al., 2017). The total loss is a weighted sum of 9 different losses related to orientation, segmentation mask, and minutiae. © IEEE. Reprinted, with permission, from Tang et al. (2017)



third maps are minutiae X/Y probability maps, which are used for precise localization by an 8-discrete location classification task. Finally, the last map is a minutiae direction distribution map, which represents the minutiae direction and is similar to the orientation distribution map.

The loss functions for each map are shown in Fig. 6.25. Minutiae manually marked by latent examiners were used as ground truth minutiae. Since there is no ground truth for orientation and segmentation, weak and strong labels are generated by ground truth minutiae and mated tenprint fingerprints. The weak orientation labels are orientation fields of aligned tenprint fingerprints extracted by traditional methods. The strong orientation labels are unoriented minutiae directions. Finally, the weak segmentation labels of a latent are obtained by dilating the convex hull of minutiae.

The authors carried out experiments on NIST DB27 and FVC2004 databases. The mean error of location and angle between extracted minutiae and ground truth is 4.4 pixels and 5.0°, respectively, on NIST DB27. On FVC2004, the mean error of location and angle is 3.4 pixels and 6.4°, respectively. Figure 6.26 shows an example of a latent fingerprint, with orientation field, foreground area, and minutiae extracted by FingerNet.

In addition, an identification experiment was conducted to test whether fingerprint matching can benefit from FingerNet. Results showed that FingerNet outperformed other

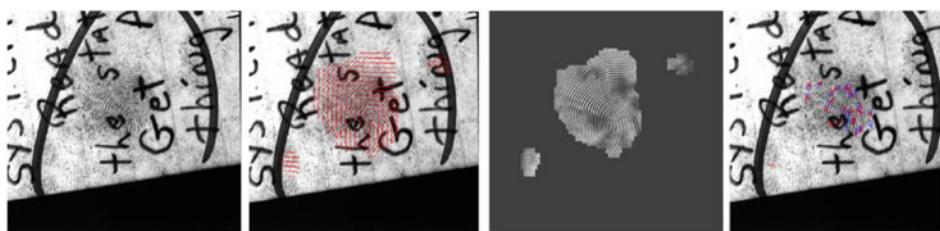


Fig. 6.26 Orientation field, foreground area, and minutiae extracted by FingerNet from a latent fingerprint (Tang et al., 2017). © IEEE. Reprinted, with permission, from Tang et al. (2017)

methods due to better minutiae extraction. The rank-1 identification rate of FingerNet is about 19% higher than the result of VeriFinger.

Some researchers have focused on extracting the ridges of latent fingerprints. Dabouei et al. (2018) proposed a direct latent fingerprint reconstruction model based on conditional generative adversarial networks (cGANs). Two modifications are applied to the cGAN to adapt it for the task of latent fingerprint reconstruction. First, the model is forced to generate three additional maps to the ridge map to ensure that the orientation and frequency information are considered in the generation process and to prevent the model from filling large missing areas and generating spurious minutiae. Second, a perceptual ID preservation approach is developed to force the generator to preserve the ID information during the reconstruction process. Using a synthetically generated database of latent fingerprints, the deep network is trained to predict missing information from the input latent samples.

The proposed model consists of three networks: a generator, a fingerprint perceptual ID information (PIDI) extractor, and a discriminator (Fig. 6.27). The generator is a “U-net” auto-encoder CNN which takes the input latent fingerprints and generates the ridge, frequency, orientation, and segmentation maps simultaneously. The reconstruction error is the weighted sum of the errors of the generated maps and their respective ground truth values. After this, generated maps are concatenated with the input latent fingerprint to provide a condition for the discriminator. Ground truth maps are extracted from the original clean fingerprints which are first distorted to simulate latent samples. During the training phase, these maps are used to provide supervision for the discriminator.

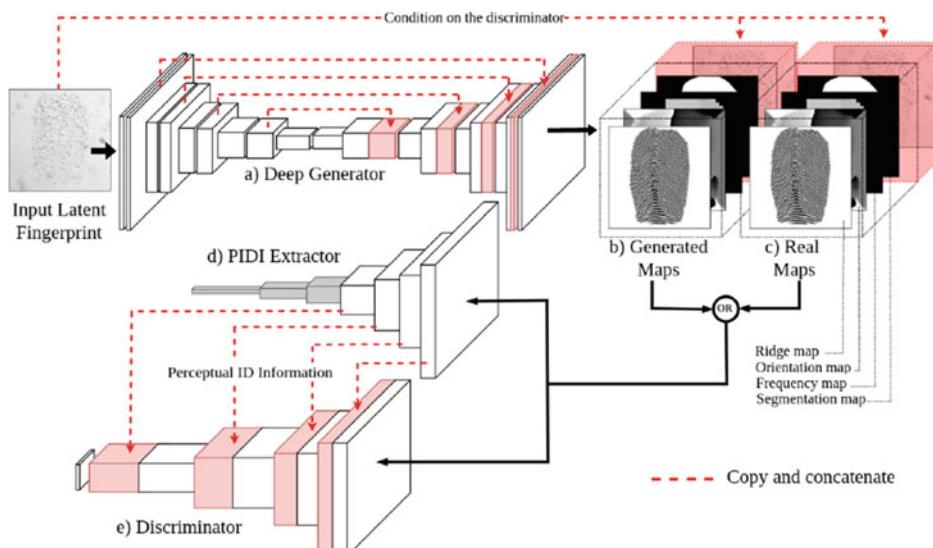


Fig. 6.27 Complete diagram of the cGAN model for latent fingerprint reconstruction (Dabouei et al., 2018). © IEEE. Reprinted, with permission, from Dabouei et al. (2018)

The fingerprint PIDI extractor is one tower from a deep Siamese fingerprint verifier that is trained using a contrastive loss. It is trained as a fingerprint verifier to extract the perceptual ID information (PIDI) of the generated maps. Extracted PIDI are output feature maps of the first four convolutional layers of the verifier module and are concatenated to the corresponding layers of the discriminator to emphasize the ID information on the discriminator's decision.

The discriminator is a deep CNN which maps the conditioned output of the generator with a size of $256 \times 256 \times 5$ to a discrimination matrix of size $16 \times 16 \times 1$. The corresponding latent fingerprint is concatenated to either the generated or the ground truth maps to act as the condition. PIDI obtained by the fingerprint verifier is also passed to the discriminator.

The authors carried out experiments on IIIT-Delhi Latent and Multi Sensor Latent Fingerprint (MOLF) database. The rank-50 accuracy for the latent-to-sensor matching on MOLF database is 70.89%, and the rank-10 accuracy for the latent-to-latent matching is 88.02%. In addition, measuring the quality of reconstructed fingerprints using NFIQ shows that the generated fingerprints are significantly enhanced compared to the raw latent samples.

6.4.7 Quality Estimation

Quality or value of a latent fingerprint is a quantitative prediction of its matching performance (Grother & Tabassi, 2007). If the mated exemplar of a latent can be identified at a high rank on a very large database, we think this latent is of high quality. Although quality estimation of fingerprint images has been already addressed in Sect. 3.11, here we focus on the specific problem of quality estimation of latent fingerprints.

Matching performance of a latent fingerprint is dependent on at least three components: the latent fingerprint, the mated exemplar fingerprint, and the matcher. Since the mated exemplar fingerprint is not available before identification, estimating the quality of a latent fingerprint is an ill-posed problem. Fortunately, exemplar fingerprints can be assumed to be of high quality since they are captured in an attended mode. Even though the quality of some exemplar fingerprints is poor, only the matching performances of their mated latent fingerprints are difficult to estimate. Therefore, the influence of exemplar fingerprints can be ignored when estimating the quality of latent fingerprints.

Matching performance of a latent fingerprint is highly dependent on the specific matcher (either an automatic matcher or a latent examiner). This is evident in performance evaluations where different matchers and different examiners demonstrate large variations in performance (Indovina et al., 2011; Ulery et al., 2011). To maximize utility (in terms of accuracy prediction), the quality should be related to a specific matcher. A latent is viewed as high quality by matcher A if its matching performance is good according to matcher A. The same latent should be assessed as low if its matching performance

by matcher B is poor. In some scenarios, it is beneficial to define quality that is not linked to a specific matcher. For example, multiple AFISs may be used to search a latent, or the AFIS may be updated, or obtaining the quality label of many latent fingerprints from a single examiner is not reliable or impractical.

While quality may have multiple applications, a major application receiving more attention is to improve efficiency without sacrificing recognition performance in a typical latent fingerprint identification workflow. If the quality of a latent is very high, “lights-out” mode can be adopted to save the time of latent examiners; if the quality is too low, the latent can be discarded; while for latent fingerprints of medium quality, feature markups should be done by examiners before submitting for AFIS search.

Traditionally, the quality of latent fingerprints is estimated by examiners. However, this has some limitations:

- Examiners quantize latent quality into only three quality levels: value for individualization (VID), value for exclusion (VEO), and no value (NV). Some consider only two levels (VID or non-VID). This ignores the fact that the quality of latent fingerprints is a continuous value, making it impossible to discriminate the matching performances of different latent fingerprints with the same quantized quality level. It will also limit the use of quality in situations where continuous quality value is desirable, such as quality-based fusion of multiple matchers.
- Quality estimation by examiners is subjective and has large variances. As reported in Ulery et al. (2011, 2012), repeatability and reproducibility of quality estimation by examiners are not high.
- Quality estimation by examiners is not very consistent with the matching performance of AFIS. The ELFT-EFS report emphasizes that a significant portion of the latent fingerprints assessed as being of VEO or NV can still be successfully identified by AFIS (Indovina et al., 2011). This is not surprising since (1) an examiner assigns a quality level to a latent based on the prediction of his/her own matching performance rather than the performance of AFIS; (2) it is difficult for an examiner to predict the performance of a specific AFIS, especially when the examiner is not familiar with it.

Understanding how latent examiners assess latent quality is important for developing automatic latent quality estimation algorithms. According to SWGFAST (2013), latent value assessments by examiners are based on the quality of features (clarity of the observed features), the quantity of features (e.g., number of minutiae and friction ridge area), the specificity of features, as well as the relationship between features in a latent. Among these four aspects, quality and quantity are usually considered in various studies, which are illustrated by four examples in Fig. 6.28.

Yoon et al. (2012) proposed a latent quality estimation approach. They first performed a feature selection procedure to select two best features: average ridge clarity in the convex hull enclosing all the minutiae manually marked (Q_R), and the number of minutiae (N_M).

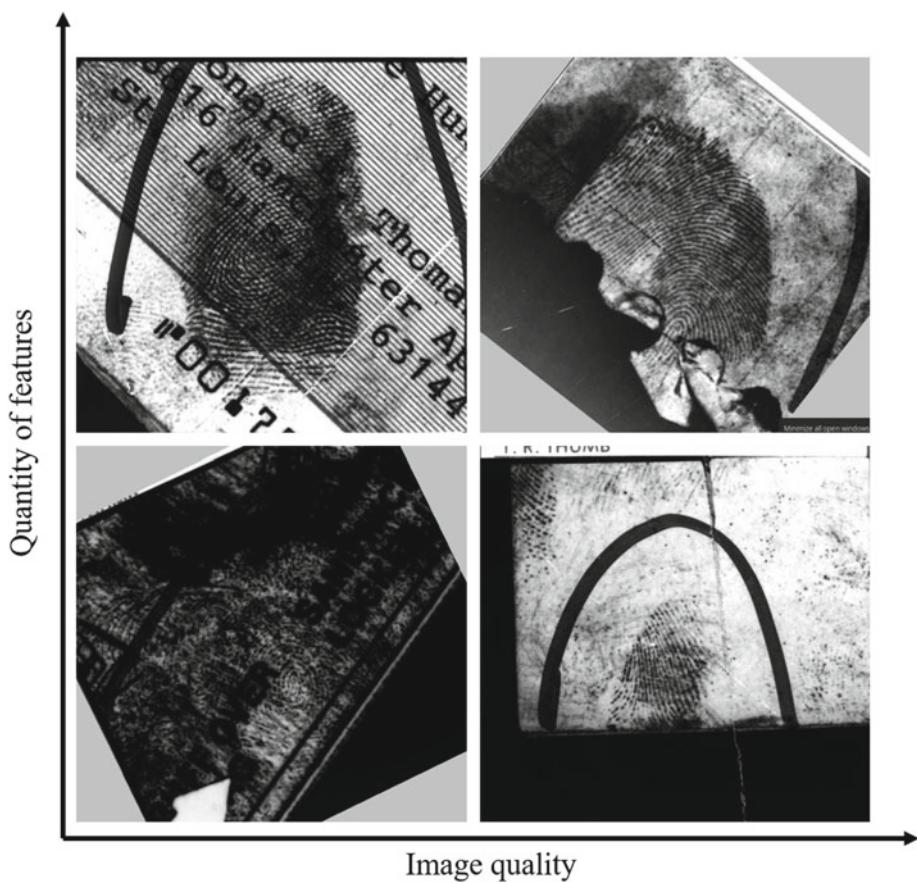


Fig. 6.28 Quality and quantity are two different dimensions of latent fingerprint value

Then a latent fingerprint image quality measure (LFIQ) is defined as $LFIQ = Q_R \cdot N_M$. For feature selection, they proposed a two-class classification problem (VID versus non-VID) and selected the features which produce the highest classification accuracy. Feature vectors with different compositions of ridge clarity features and minutiae features were evaluated in terms of the resulting classification accuracy. To evaluate the usefulness of the quality measure, latent identification experiments were performed on a dataset by combining the NIST DB27 and WVU latent datasets. LFIQ improved the rank-100 identification rate from 69 to 80% by rejecting 21% of latent fingerprints deemed as poor quality. However, the same identification rate can only be achieved by rejecting 80% of the latent fingerprints in the databases assessed as “NFIQ = 5” (Sect. 3.11.3).

Yoon et al. (2013) proposed a new LFIQ measure by considering more factors: (1) global connectivity of good ridge quality regions, (2) minutiae reliability based on a

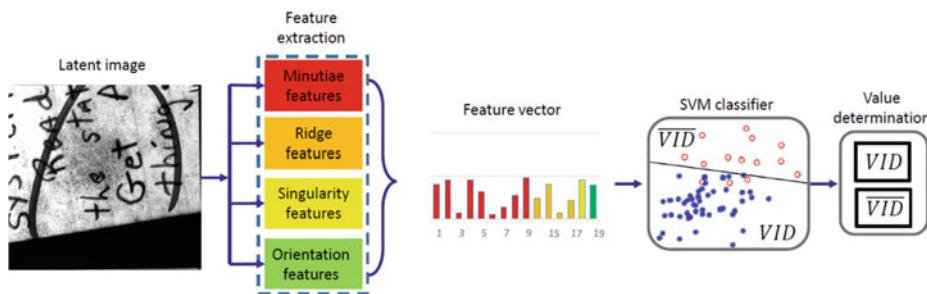


Fig. 6.29 Automatic latent value determination algorithm in Cao et al. (2016). © IEEE. Reprinted, with permission, from Cao et al. (2016)

minutiae dictionary learnt from high-quality minutiae patches, and (3) finger position by detecting a reference point. The new LFIQ metric is defined based on the triangulation of minutiae incorporating the above three factors. A combination of manually marked minutiae and automatically extracted minutiae was used. Identification experiments showed that this new LFIQ measure leads to higher identification rates by rejecting the same number of low-quality latent fingerprints.

Cao et al. (2016) proposed a fully automatic method for latent value determination based on the number, reliability, and compactness of the minutiae, ridge quality, ridge flow, and the number of core and delta points. Given the small number of latent fingerprints with VEO and NV labels, only a two-class value determination is considered, namely VID and non-VID (VEO and NV). An SVM was used to classify a 19-D feature vector extracted from a latent as VID or non-VID (see Fig. 6.29). Experiments showed that the performance of this fully automatic quality assessment method is comparable to LFIQ in Yoon et al. (2013), which requires manually marked minutiae.

Chugh et al. (2018) proposed a crowdsourcing-based framework for understanding the underlying bases of value assignment by fingerprint examiners and used it to learn a predictor for quantitative latent value assignment. A comparison between the correlation values suggested that the proposed latent value predictor is better than LFIQ and the value by Cao et al. (2016) in predicting the AFIS performance.

Limited by the small size of public latent fingerprint datasets, published latent quality estimation approaches typically used manually designed features and computed quality levels by empirically designed formulas or by trained traditional classifiers/regressors. Large training datasets are necessary for learning the complex mapping from latent image to AFIS matching performance.

6.5 Matching

6.5.1 Challenges

For fingerprint matching algorithm, an ideal fingerprint image should have a large fingerprint area, high-quality ridge pattern, and negligible skin distortion. However, in the case of latent fingerprints, the opposite of these requirements is often the case. This makes latent fingerprint matching a very challenging problem. Here we compare latent fingerprints with plain/rolled fingerprints through some statistical analysis.

Latent fingerprints usually have a small finger area as compared to rolled prints. Due to the small area, the number of minutiae may be very small, e.g., ≤ 10 . Figure 6.30 shows the distributions of the number of minutiae of latent and rolled fingerprints in NIST DB27 database and plain fingerprints in FVC2002 DB1. Minutiae were manually marked in latent fingerprints and automatically extracted in rolled and plain fingerprints. Given this small number of minutiae, minutiae alone contain very limited information for identification.

Due to the presence of background noise (characters and strokes on many fingerprints scanned from paper), the feature extraction techniques may miss true minutiae and produce many spurious minutiae, which further decrease the consistency of minutiae descriptors such as MCC (Sect. 4.4.3). Figure 6.31 shows the distributions of the numbers of paired minutiae in latent-to-rolled comparisons on NIST DB27 latent database, in rolled-to-rolled comparisons on NIST DB4 database, and in plain-to-plain comparisons on FVC2002 DB1. Minutiae were manually marked in latent fingerprints and automatically extracted in rolled and plain fingerprints. The number of paired minutiae estimated by the MCC descriptors and a spectral graph matching algorithm (Sect. 4.3.3) when comparing latent-to-rolled fingerprints is far smaller than the other two cases.

Fig. 6.30 Distributions of the number of minutiae in latent and rolled fingerprints in NIST DB27 database and plain fingerprints in FVC2002 DB1

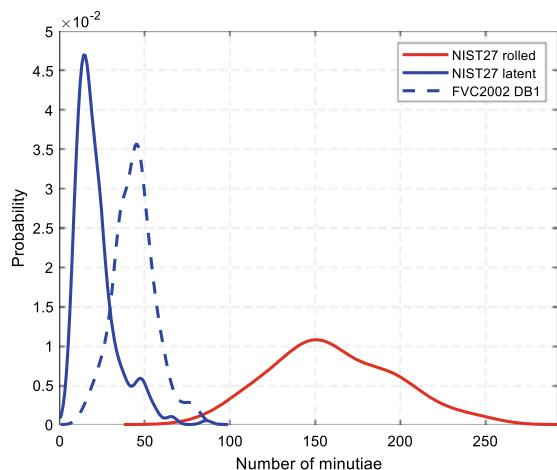
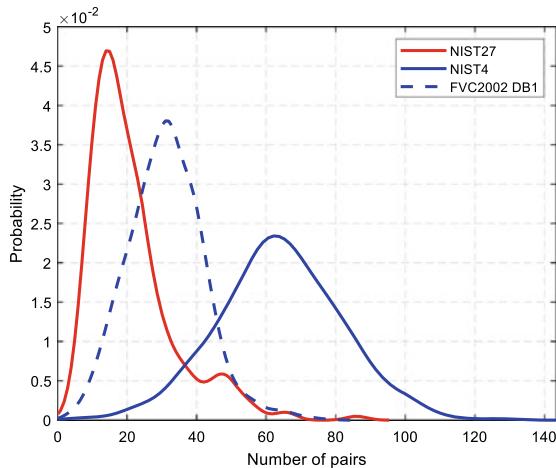


Fig. 6.31 Distributions of the number of paired minutiae estimated by an automatic minutiae matcher on the NIST DB27 and NIST DB4 databases, respectively



Latent prints collected from crime scenes or items of evidence may exhibit a myriad of distortion issues (Maceo, 2009). The elastic distortion introduced due to the inherent elasticity of human skin, the contact-based fingerprint acquisition procedure, and a purposely lateral force or torque, may increase the intra-class variations (difference among fingerprints from the same finger) and thus lead to false non-matches due to limited capability of existing fingerprint matchers in recognizing severely distorted fingerprints. From Fig. 6.32, it is evident that the distortion on latent fingerprints in the NIST DB27 database is more serious than that on plain fingerprints in the FVC2002 DB1 database. After rigid registration, the average location error of paired minutiae is 7.5 pixels on NIST

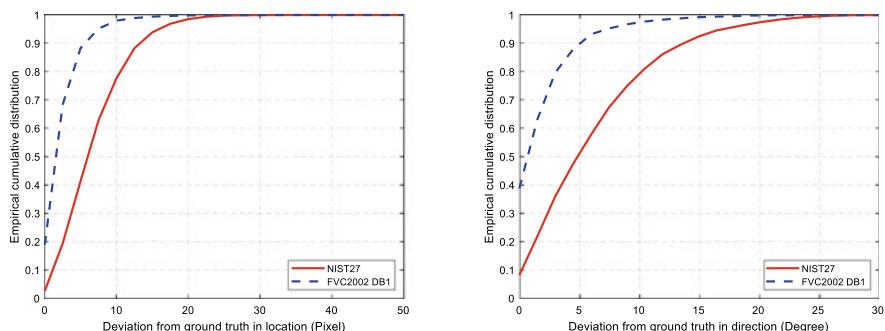


Fig. 6.32 Cumulative error distribution of the distance and angle difference between mated minutiae point pairs after rigid registration. To ensure that the deformation is calculated according to the correct minutiae pairing, latent-to-rolled registration on NIST DB27 is based on manually matched minutiae by latent examiners. Fingerprint registration on FVC2002 DB1 is performed based on automatically matched minutiae

DB27 database, and 4.2 pixels on FVC2002 DB1 database. A pair of latent and rolled fingerprints with large distortion is shown in Fig. 6.33.

Due to the above difficulties in latent fingerprint matching, fingerprint matchers designed for plain/rolled fingerprint matching usually perform much worse in latent matching. Figure 6.34 shows the DET curves of VeriFinger SDK 12.0 on NIST DB27

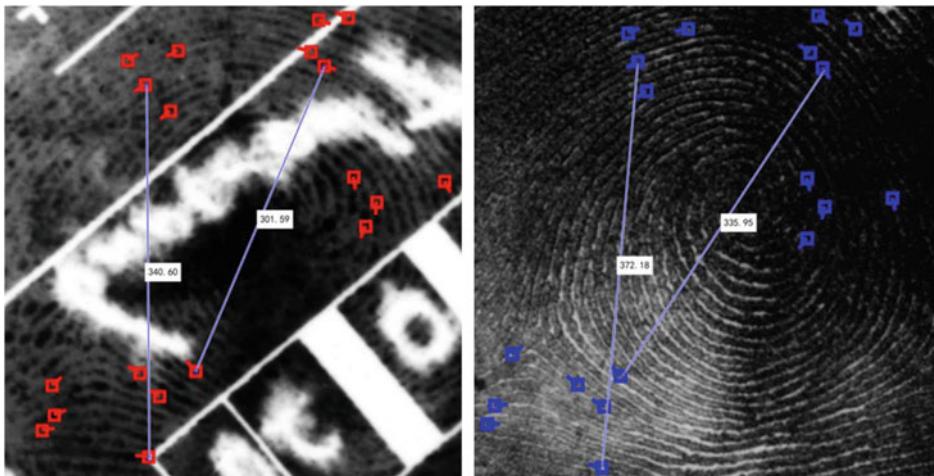
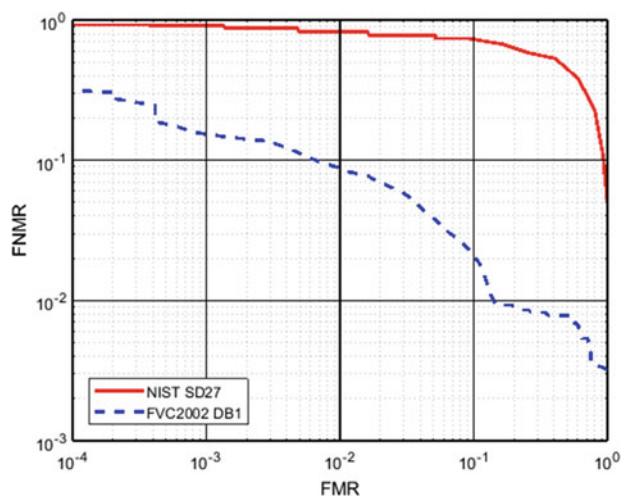


Fig. 6.33 The distortion between a latent fingerprint and its mated rolled fingerprint can be very large. In this example, two pairs of mated minutiae are connected and their distances are labeled on the images. The differences are more than 30 pixels

Fig. 6.34 DET curves of VeriFinger SDK 12.0 on NIST DB27 and FVC2002 DB1. Verification mode is used to make the matching performances on two databases comparable



and FVC2002 DB1. To improve the performance of latent matching, the aforementioned challenges have to be taken into consideration when designing latent matching algorithms.

6.5.2 Latent Matching with Manually Marked Features

Due to the lack of a robust feature extractor for latent fingerprints, early latent fingerprint matching studies mainly used manually marked features in latent fingerprints (Jain & Feng, 2011; Paulino et al., 2012).

Jain and Feng (2011) proposed a latent-to-rolled matching algorithm that utilizes multiple fingerprint features including Level-1 features (ridge flow map, ridge wavelength map, core, delta), Level-2 features (minutiae, ridge skeleton image), and ridge quality map. Multiple features are chosen to improve the representation capability of fingerprint templates. These features are manually marked for latent fingerprints, but they are automatically extracted for rolled prints and the matching algorithm is also automatic. A rank-1 identification rate of 74% was obtained in matching latent images in NIST DB27 against a background database consisting of NIST DB27, DB4, and DB14. In order to evaluate the relative importance of each extended feature, these features were incrementally used in the order of their cost in marking by latent examiners. The experimental results showed that singularities, ridge quality maps, and ridge flow maps were the most effective features in improving the matching accuracy.

Alignment is the first step in latent fingerprint matching and hence is critical for the matching performance. Paulino et al. (2012) proposed an alignment method based on MCC descriptors and a Generalized Hough Transform. Given two minutiae sets from latent and rolled fingerprints, respectively, a matching score is computed by the similarity between the local MCC descriptors for each possible transformation in a discretized set. The matching score is further used as a vote on the corresponding transformation, and the transformation receiving the most votes is chosen. After that, the matching score between two aligned fingerprints is computed by the similarity between the minutiae sets and orientation fields. Better performance is shown compared with three state-of-the-art fingerprint matchers for NIST DB27 and WVU latent databases against a background database of around 32 K rolled prints.

Arora et al. (2014) used rolled fingerprints as feedback in latent fingerprint matching. After initial matching and alignment, a matching score is obtained, and the latent fingerprint is aligned to the rolled one. Then the features of the latent fingerprint are refined based on the feedback from features of corresponding rolled ones, and the feedback feature similarity is accordingly updated. The final matching score is a fusion of the initial and the feedback matching scores. A marked improvement in the identification accuracy was observed on NIST DB27 and WVU latent databases against a background database of 100 K rolled prints when using the feedback.

Krish et al. (2015) applied the orientation field to register the latent fingerprints with rolled fingerprints in two hierarchical levels. In the first level, the normalized correlation between the orientation field of a latent fingerprint for each rotation parameter and that of a rolled fingerprint is computed, and the correlation peaks are taken as candidate registration locations. In the second level, multiple similarities are calculated on these candidate locations, and the candidate with the largest similarity is chosen as the final registration location. With the pre-registration, the search space of minutiae is reduced, which helps improve the performance of minutiae matchers.

Krish et al. (2019) studied incorporating rare minutiae including fragments, enclosures, dots, interruptions, and so on to improve latent fingerprint identification accuracy. With rare minutiae as reference points, the least square fitting error is estimated for an affine transformation between latent and rolled minutiae sets. The similarity scores of minutiae-based matchers are further modified based on the fitting error. Improvement in the rank identification accuracies is observed when applying rare minutiae features to modify the similarity scores of three widely used minutiae matchers.

Feng et al. (2009) studied the fusion of plain and rolled fingerprints to improve the matching performance of latent fingerprints. Multiple fusion approaches are proposed at three different levels: rank-level, score-level, and feature-level. Among them, most fusion approaches improve the identification performance in searching 230 latent fingerprints in the ELFT-EFS Public Challenge Dataset against an exemplar database of 4180 pairs of rolled and plain fingerprints, and the boosted max score level fusion shows the greatest improvement.

6.5.3 Latent Matching with Automatically Extracted Features

With the development of latent fingerprint feature extraction technology, especially deep learning-based methods, latent fingerprint matching based on automatically extracted features is becoming possible. Cao and Jain (2019) made three innovations in this direction: ConvNet-based minutiae descriptor, multiple minutiae templates, and virtual minutiae.

Minutiae descriptor is a very important component of minutiae matching. In the past, minutiae descriptors were usually designed by experience. Among them, well-designed descriptors (such as MCC) perform quite well in the matching of livescan and inked fingerprints. However, in latent fingerprint matching, the performance of these descriptors decreases greatly due to the lack of minutiae and the low reliability of automatically extracted minutiae. Cao and Jain (2019) proposed to use ConvNet for minutiae descriptor extraction.

The minutiae descriptor is learnt from multiple image patches (a total of 14 patches) at different scales and locations, as illustrated in Fig. 6.35. For each image patch extracted from the same minutia, one ConvNet is trained to obtain the feature vector, and finally,

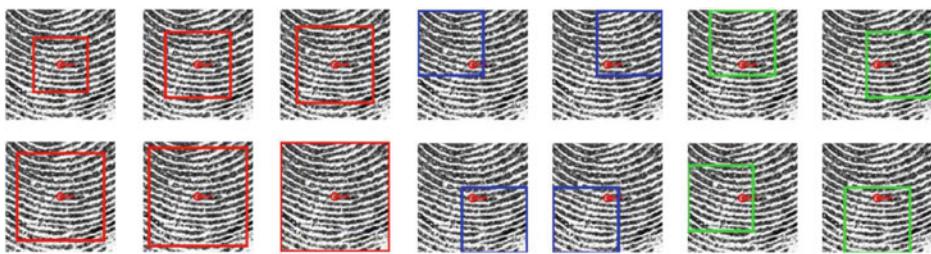


Fig. 6.35 Multiple image patches at different scales and locations for computing descriptor of a minutia (Cao & Jain, 2019). © IEEE. Reprinted, with permission, from Cao and Jain (2019)

a subset of the 14 feature vectors output by the 14 ConvNets are concatenated into a minutia descriptor.

The training minutiae patches are extracted from fingerprints in the Michigan State Police longitudinal fingerprint database, which contains 1311 subjects with at least 10 rolled impressions for each finger. Each minutia is considered as a class, and only classes with more than 8 image patches are retained. In this case, each ConvNet is trained as a multi-class classifier. At test time, the output of the last fully connected layer of each ConvNet is considered as the feature vector of the input image patch.

Considering that a single feature extraction algorithm may not be reliable enough for latent fingerprints, two different algorithms are used for feature extraction. They are based on different orientation field estimation methods (ConvNet-based and Dictionary-based) and ridge enhancement methods (Dictionary-based and Gabor filtering-based). Minutiae descriptors are extracted by ConvNets using local latent patches obtained by two feature extraction methods. See Fig. 6.36 for the flowchart of this approach.

Besides, they proposed a texture template to account for situations where the latent is of such a small area that it does not contain a sufficient number of minutiae (for reliable comparison to exemplar prints) or the latent is of such poor quality that minutiae extraction is not reliable. In a texture template, they represent each non-overlapping local block (16×16 pixels) in the latent by a pair of virtual minutiae. Let (x, y) and θ be the location and ridge orientation of the center of a block. Then the virtual minutiae pair is located at (x, y, θ) and $(x, y, \theta + \pi)$. Note that the virtual minutiae do not correspond to ridge endings and bifurcations, and the virtual minutiae close to the border are removed. The same minutia descriptor algorithm used for the true minutiae sets is also used for virtual minutiae.

When comparing latent fingerprints to rolled fingerprints, they first apply the second-order and third-order graph matching algorithms to establish the minutiae correspondences (see Graph matching algorithms in Sect. 4.3.3). The same minutiae comparison algorithm was also used for virtual minutiae comparison in texture template. The final similarity score between the latent and the exemplar print is computed as the weighted sum of

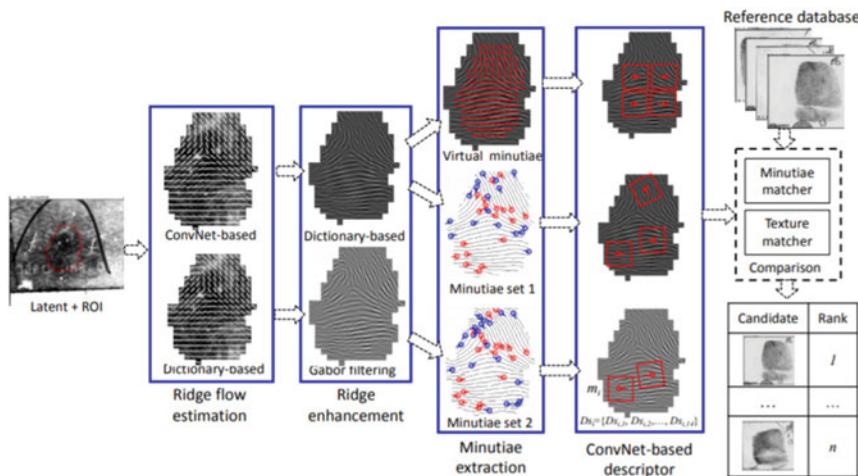


Fig. 6.36 Flowchart of the latent fingerprint recognition approach in Cao and Jain (2019). © IEEE. Reprinted, with permission, from Cao and Jain (2019)

two minutiae template similarity scores generated by comparing the two latent minutiae templates against the single exemplar minutiae template and the texture similarity score generated by comparing the latent and exemplar print texture templates. For exemplar fingerprints, they extract only one minutiae template and texture template since exemplar prints are typically of higher quality compared to latent fingerprints.

Experimental results on two latent databases, NIST DB27 and WVU, showed that the recognition performance of the virtual minutiae-based texture template, when fused with two different true minutiae templates, boosted the rank-1 accuracy from 58.5% to 64.7% against a 100 K exemplar print gallery for NIST DB27.

Later, Cao et al. (2020) built a fully automated end-to-end system and improved the search accuracy and computational efficiency of the system based on the work of Cao and Jain (2019). The work in Cao and Jain (2019) has mainly the following limitations: (i) manually marked ROI is needed, (ii) skeleton-based minutiae extraction introduces a large number of spurious minutiae, and (iii) a large texture template size (1.4 MB) makes latent-to-exemplar comparison extremely slow. Therefore, Cao et al. (2020) improved both identification accuracy and search speed of texture templates by (i) reducing the template size, (ii) improving the efficiency of graph matching, and (iii) implementing the matching code across multiple threads in C++.

Virtual minutiae in Cao and Jain (2019) still suffer from unstable estimation of local ridge orientation, which is common in latent fingerprints. In addition, virtual minutiae are not salient features and cannot be located accurately. In order to overcome the limitations of virtual minutiae, Gu et al. (2021) proposed a latent fingerprint registration algorithm based on dense sampling points. A fingerprint is represented as dense sampling points

to avoid the minutiae extraction step and ensure the adequacy of key points even if the fingerprint area is very small. Then the local patch alignment and matching provides an estimate for the alignment parameters between image patches and computes their similarities. The estimation of relative rotation between two sampling points rather than a separate estimation of their own local direction helps obtain more accurate alignment. The estimation of relative offset between two sampling points helps achieve pixel-level alignment accuracy without the need of sampling every pixel. In addition, the local patch matching module computes the similarity of two original image patches rather than enhanced ones to ensure that the image will not be damaged by the fingerprint enhancement algorithm. Experimental results on NIST DB27 showed that this approach performed better than previous methods, especially under challenging conditions such as severe background noise and small fingerprint area.

6.5.4 Performance Evaluation

Performance evaluation of fingerprint recognition algorithms on common databases is important for finding the strengths and weaknesses of state-of-the-art technologies. Currently, NIST DB27 is the main test database used in scientific papers for latent fingerprints, followed by MOLF (Sankaran et al., 2015) and the WVU latent fingerprint database. However, the size of these databases is small and only NIST DB27 is collected from crime scenes. Unfortunately, NIST DB27 was recently revoked along with other NIST fingerprint databases.

NIST has conducted a series of tests on latent fingerprint identification technology, including Evaluation of Latent Fingerprint Technologies (ELFT) and Evaluation of Latent Fingerprint Technologies-Extended Feature Sets (ELFT-EFS). Participants are mainly AFIS vendors.

In the latest one, ELFT-EFS Evaluation #2 (Indovina et al., 2012), the purpose is to evaluate the state of the art in latent feature-based matching, by comparing the accuracy of searches using images alone with searches using different sets of features marked by experienced latent examiners. The dataset contains 1066 latent fingerprints from 826 subjects, with feature markup by latent examiners. The gallery contains all ten rolled and plain exemplar fingerprints from these 826 subjects and 99,163 other subjects.

The features include minutiae, ridge counts, cores, deltas, pattern class, ridge quality maps, ridge skeletons, ridge flow maps, creases, dots, incipient ridges, ridge edge protrusions, and pores. Latent examiners made determinations of Value, Limited Value (latent fingerprints of value for exclusion only), or No Value at the time of markup, in addition to subjective quality assessments of “Excellent”, “Good”, “Bad”, “Ugly”, and “Unusable”. Features were marked in latent images without reference to exemplar fingerprints, with the exception of a subset of 418 latent images that included an additional Ground Truth (GT) markup based on the latent and all available exemplars; GT markup provides a measure of

ideal (but operationally infeasible) performance when compared to the original examiner markup.

Some of the key findings in this test are summarized below:

- Image-only searches were more accurate than feature-only searches for all matchers.
- Most fingerprint matchers benefited from manually marked features when provided along with the latent image.
- Using additional EFS features other than minutiae results in accuracy improvement.
- The highest measured accuracy achieved by any individual matcher at rank 1 on any latent feature subset (excluding the use of ground truth markup) was 71.4%. This indicates a potential for additional accuracy improvement through improved algorithms. The differences in which latent fingerprints were identified by the various matchers also points to a potential accuracy improvement by using algorithm fusion.
- The performance of all matchers decreased consistently as lower quality latent fingerprints were searched, with respect to the subjective quality assessment.
- Analysis showed that the greatest percentage of the misses were for latent fingerprints with low minutiae count, and those assessed by examiners as poor quality. Algorithm accuracy for all participants was highly correlated to the number of minutiae.
- The ground truth (GT) markup method, in which all exemplar mate images were consulted when marking latent features, yielded an increase in performance over the original examiner markup of about 4 to 6 percentage points for image + full Extended Feature Set (EFS) searches, and about 12 to 15 percentage points for minutiae-only searches. Though this method is obviously not practical in an operational scenario, however, it shows that matcher accuracy is highly affected by the precision of latent examiner markup, especially in the absence of image data.

However, no participating algorithms in this test are published and the databases are not publicly available. Moreover, the latest assessment by NIST was in 2012 and did not reflect the latest advances in deep learning technology. In May 2020, NIST announced the start of a new round of latent fingerprint technology evaluation to cover all types of friction ridges, including palms, which may reflect the latest advances in technology.

6.6 Summary

Latent fingerprint recognition is challenging due to poor quality, small area, large deformation, and other factors. However, latent fingerprint recognition is a very important research topic, not only because of its irreplaceable role in identifying suspects, but also because the study of latent fingerprint can produce technologies that are also beneficial to other fingerprint recognition problems.

Great progress has been made in this field in recent years. A significant improvement is that deep learning technology has achieved better performance than traditional methods in several of the main steps of fingerprint identification. Despite this progress, latent search using features marked by examiners is still the major type of latent search. For example, according to the fact sheet of the FBI's NGI system in June 2021 (FBI, 2021), there were 5,031 image searches and 20,166 feature searches. For a definitive transition to full “lights-out” mode (Meagher & Dvornychenko, 2011), further technical improvement is necessary.

There is some room for improvement:

- The lack of public fingerprint database is a big obstacle to algorithm research. At present, deep learning has become the main technology in latent fingerprint identification, which requires a large number of samples. NIST DB27 is far too small for deep network training and testing, especially considering high diversity of latent fingerprints. In addition, this database is no longer available on the official website.
- Exemplar-to-latent and latent-to-latent identification have not been further analyzed. Although not fundamentally different from the latent-to-exemplar matching, the study of these problems should further stimulate technological progress.
- A latent fingerprint recognition algorithm includes several steps. It is important to evaluate each step independently to identify problems and to develop better methods. However, only orientation field estimation and fingerprint region segmentation have independent evaluation.
- Compared with feature extraction, matching and scoring have not yet been revolutionized by deep learning. In particular, in most studies, different matching algorithms and scoring methods are not decoupled from feature extraction, making it difficult to evaluate different matching algorithms and scoring methods.

References

- Arora, S.S., Liu, E., Cao, K., & Jain, A.K. (2014). Latent fingerprint matching: Performance gain via feedback from exemplar prints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(12), 2452–2465.
- Arora, S.S., Cao, K., Jain, A.K., & Michaud, G. (2015). Crowd powered latent fingerprint identification: Fusing AFIS with examiner markups. In *Proceedings of International Conference on Biometrics* (pp. 363–370).
- Blake, A., Kohli, P., & Rother, C. (2011). *Markov random fields for vision and image processing*. MIT Press.
- Budowle, B., Buscaglia, J., & Perlman, R. S. (2006). Review of the scientific basis for friction ridge comparisons as a means of identification: Committee findings and recommendations. *Forensic Science Communications*, 8(1), 1–16.

- Cao, K., Liu, E., & Jain, A. K. (2014). Segmentation and enhancement of latent fingerprints: A coarse to fine ridgestructure dictionary. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(9), 1847–1859.
- Cao, K., Chugh, T., Zhou, J., Tabassi, E., & Jain, A. K. (2016). Automatic latent value determination. In *Proceedings of International Conference on Biometrics* (pp. 1–8).
- Cao, K., & Jain, A. K. (2015). Latent orientation field estimation via convolutional neural network. In *Proceedings of International Conference on Biometrics* (pp. 349–356).
- Cao, K., & Jain, A. K. (2019). Automated latent fingerprint recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(4), 788–800.
- Cao, K., Nguyen, D. L., Tymoszek, C., & Jain, A. K. (2020). End-to-end latent fingerprint search. *IEEE Transactions on Information Forensics and Security*, 15, 880–894.
- Champod, C. (2015). Fingerprint identification: Advances since the 2009 National Research Council report. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 370(1674). Article ID: 20140259.
- Champod, C., Lennard, C. J., Margot, P., & Stoilovic, M. (2017). *Fingerprints and other ridge skin impressions* (2nd), CRC Press.
- Chen, F., Feng, J., Jain, A. K., Zhou, J., & Zhang, J. (2011). Separating overlapped fingerprints. *IEEE Transactions on Information Forensics and Security*, 6(2), 346–359.
- Chugh, T., Cao, K., Zhou, J., Tabassi, E., & Jain, A. K. (2018). Latent fingerprint value prediction: Crowd-based learning. *IEEE Transactions on Information Forensics and Security*, 13(1), 20–34.
- Cole, S. A. (2005). More than zero: Accounting for error in latent fingerprint identification. *Journal of Criminal Law & Criminology*, 95(3), 985–1078.
- Dabouei, A., Kazemi, H., Iranmanesh, S. M., Dawson, J., & Nasrabadi, N. M. (2018). ID preserving generative adversarial network for partial latent fingerprint reconstruction. In *Proceedings of International Conference on Biometrics Theory, Application and Systems* (pp. 1–10).
- Dror, I.E., Charlton, D., & Péron, A. E. (2006). Contextual information renders experts vulnerable to making erroneous identifications. *Forensic Science International*, 156(1), 74–78.
- Duan, Y., Feng, J., Lu, J., & Zhou, J. (2021). Orientation field estimation for latent fingerprints with prior knowledge of fingerprint pattern. In *Proceedings of International Joint Conference on Biometrics* (pp. 1–8).
- FBI (2021). FBI Biometric services section, June 2021 Next Generation Identification (NGI) system fact sheet. <https://www.fbi.gov/file-repository/ngi-monthly-fact-sheet>. Accessed Jul 2021.
- Feng, J., Yoon, S., & Jain, A. K. (2009). Latent fingerprint matching: Fusion of rolled and plain fingerprints. In *Proceedings of International Conference on Biometrics* (pp. 695–704).
- Feng, J., Shi, Y., & Zhou, J. (2012). Robust and efficient algorithms for separating latent overlapped fingerprints. *IEEE Transactions on Information Forensics and Security*, 7(5), 1498–1510.
- Feng, J., Zhou, J., Jain, A. K. (2013). Orientation field estimation for latent fingerprint enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(4), 925–940.
- Gische, M. R. (2012). Recent trends in fingerprint evidence. *Texas Forensic Science Seminar*. http://www.txcourts.gov/All_Archived_Documents/ccaInformation/tcjju/ppt/Gische-9.pptx. Accessed Jul 2021.
- Grother, P., & Tabassi, E. (2007). Performance of biometric quality measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4), 531–543.
- Gu, S., Feng, J., Lu, J., & Zhou, J. (2021). Latent fingerprint registration via matching densely sampled points. *IEEE Transactions on Information Forensics and Security*, 16, 1231–1244.
- Hicklin, A. (2017). Improving the rigor of the latent print examination process, Ph.D. thesis, University of Lausanne.
- Indovina, M. D., Hicklin, R. A., & Kiebuzinski, G. I. (2011). NIST evaluation of latent fingerprint technologies: Extended feature sets [Evaluation# 1], NIST-IR 7775.

- Indovina, M. D., Dvornychenko, V., Hicklin, R. A., & Kiebuzinski, G. I. (2012). NIST evaluation of latent fingerprint technologies: Extended feature sets [Evaluation# 2], NIST-IR 7859.
- Jain, A. K., & Feng, J. (2011). Latent fingerprint matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1), 88–100.
- Kassin, S. M., Dror, I. E., & Kukucka, J. (2013). The forensic confirmation bias: Problems, perspectives, and proposed solutions. *Journal of Applied Research in Memory and Cognition*, 2(1), 42–52.
- Krish, R. P., Fierrez, J., Ramos, D., Ortega-Garcia, J., & Bigun, J. (2015). Pre-registration of latent fingerprints based on orientation field. *IET Biometrics*, 4(2), 42–52.
- Krish, R. P., Fierrez, J., Ramos, D., Alonso-Fernandez, F., & Bigun, J. (2019). Improving automated latent fingerprint identification using extended minutia types. *Information Fusion*, 50, 9–19.
- Maceo, A. V. (2009). Qualitative assessment of skin deformation: A pilot study. *Journal of Forensic Identification*, 59(4), 390–440.
- Meagher, S., & Dvornychenko, V. (2011). Defining AFIS latent print ‘lights-out’, NIST-IR 7811.
- Nguyen, D. L., Cao, K., & Jain, A. K. (2018). Automatic latent fingerprint segmentation. In *Proceedings of International Conference on Biometrics Theory, Applications and Systems*, (pp. 1–9).
- NRC (2009). National Research Council, “Strengthening forensic science in the United States: A path forward,” *National Academies Press*.
- OIG (2006). Office of the Inspector General, A Review of the FBI’s Handling of the Brandon Mayfield Case.
- Paulino, A. A., Feng, J., & Jain, A. K. (2012). Latent fingerprint matching using descriptor-based hough transform. *IEEE Transactions on Information Forensics and Security*, 8(1), 31–45.
- Sankaran, A., Vatsa, M., & Singh, R. (2015). Multisensor optical and latent fingerprint database. *IEEE Access*, 3, 653–665.
- SWGFAST (2013). Standards for examining friction ridge impressions and resulting conclusions (latent/tenprint) v2.0. http://clpex.com/swgfast/documents/examinations-conclusions/130_427_Examinations-Conclusions_2.0.pdf. Accessed Jul 2021
- Tang, Y., Gao, F., Feng, J., & Liu, Y. (2017). FingerNet: An unified deep network for fingerprint minutiae extraction. In *Proceedings of International Joint Conference on Biometrics* (pp. 108–116).
- Turroni, F., Maltoni, D., Cappelli, R., & Maio, D. (2011). Improving fingerprint orientation extraction. *IEEE Transactions on Information Forensics and Security*, 6(3), 1002–1013.
- Ulery, B. T., Hicklin, R. A., Buscaglia, J., & Roberts, M. A. (2011). Accuracy and reliability of forensic latent fingerprint decisions. *Proceedings of the National Academy of Sciences*, 108(19), 7733–7738.
- Ulery, B. T., Hicklin, R. A., Buscaglia, J., & Roberts, M. A. (2012). Repeatability and reproducibility of decisions by latent fingerprint examiners. *PloS one*, 7(3), e32800.
- Ulery, B. T., Hicklin, R. A., Kiebuzinski, G. I., Roberts, M. A., & Buscaglia, J. (2013). Understanding the sufficiency of information for latent fingerprint value determinations. *Forensic Science International*, 230(1–3), 99–106.
- Ulery, B. T., Hicklin, R. A., Roberts, M. A., & Buscaglia, J. (2015). Changes in latent fingerprint examiners’ markup between analysis and comparison. *Forensic Science International*, 247, 54–61.
- Ulery, B. T., Hicklin, R. A., Roberts, M. A., & Buscaglia, J. (2016). Interexaminer variation of minutia markup on latent fingerprints. *Forensic Science International*, 264, 89–99.
- Ulery, B. T., Hicklin, R. A., Roberts, M. A., & Buscaglia, J. (2017). Factors associated with latent fingerprint exclusion determinations. *Forensic Science International*, 275, 65–75.
- Watson, C. I., Garris, M. D., Tabassi, E., Wilson, C. L., McCabe, R. M., Janet, S., & Ko, K. (2007). User’s guide to NIST biometric image software (NBIS), NIST-IR 7392.

- Watson, C. I., Fiumara, G., Tabassi, E., Cheng, S. L., Flanagan, P., & Salamon, W. (2012). Finger- print vendor technology evaluation, NIST-IR 8034.
- Yang, X., Feng, J., & Zhou, J. (2014). Localized dictionaries based orientation field estimation for latent fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(5), 955– 969.
- Yang, X., Feng, J., Zhou, J., Xia, S. (2015). Detection and segmentation of latent fingerprints. In *International Workshop on Information Forensics and Security* (pp. 1–6).
- Yin, Q., Feng, J., Lu, J., & Zhou, J. (2018). Orientation field estimation for latent fingerprints by exhaustive search of large database. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems* (pp. 1–9).
- Yoon, S., Liu, E., & Jain, A. K. (2012). On latent fingerprint image quality. In *Proceedings of International Workshop on Computational Forensics* (pp. 67–82).
- Yoon, S., Cao, K., Liu, E., & Jain, A. K. (2013). LFIQ: Latent fingerprint image quality. In *Proceed- ings of International Conference on Biometrics Theory, Applications and Systems* (pp. 1–8).
- Zhao, Q., & Jain, A. K. (2012). Model based separation of overlapping latent fingerprints. *IEEE Transactions on Information Forensics and Security*, 7(3), 904–918.



Fingerprint Synthesis

7

Abstract

Synthetic fingerprints, when properly generated, represent a reasonable substitute for real fingerprints for the design, training, and benchmarking of fingerprint recognition algorithms. This approach is particularly useful to deal with emerging privacy regulations (e.g., EU-GDPR) limiting the use of personally identifiable information. This chapter introduces fingerprint synthesis and focuses on the two main categories of generation approaches: (i) first generate a master fingerprint and then derive multiple impressions (e.g., SFinGe); (ii) generative models (e.g., GAN) for the direct synthesis of fingerprint images. Validation of synthetic generators through large scale experiments is finally presented.

Keywords

Fingerprint synthesis • Synthetic fingerprints • Master fingerprint • Generative models • Background generation • SFinGe

7.1 Introduction

Significant efforts are continuously being made in designing new fingerprint recognition algorithms both in academic and industrial institutions. However, the accuracy of each algorithm is usually evaluated on relatively small databases. An evaluation on small databases makes the accuracy-estimates highly data dependent; as a result, they do not generalize well on fingerprint images captured in different applications and different

Invited Chapter by Raffaele Cappelli, University of Bologna.

environments. Furthermore, when the databases are proprietary, the accuracy of various matching algorithms cannot be compared directly. A sharable large database of fingerprints (tens of thousands of images) is required to evaluate and compare various fingerprint recognition algorithms, due to the very small error rates that have to be estimated. Unfortunately, collecting large databases of fingerprint images is (i) expensive both in terms of money and time; (ii) tedious for both the data collection technicians and for the subjects providing the data; (iii) critical due to sensitive data protection laws. Even if one is able to collect such a large fingerprint database, data protection laws make it difficult, if not impossible, to share it with others. Finally, publicly available databases of real fingerprints, such as those used in FVC technology evaluations (Maio et al., 2002a, b, 2004), Cappelli et al. (2006), do not constitute lasting solutions for evaluating and comparing different algorithms because they expire once “used,” and new databases have to be collected for future evaluations. In other words, once an evaluation database is released, algorithm developers can “train” their algorithm to perform well on that specific database.

A potential alternative to collecting large fingerprint databases is fingerprint sample synthesis, i.e., generating images similar to human fingerprints, through parametric models that encode the salient characteristics of such images and their modes of variation (see Buettner & Orlans, 2005; Yanushkevich et al., 2005). Existing studies on fingerprint synthesis can be grouped into two main categories:

- methods that first generate a *master fingerprint* and then derive synthetic impressions from it. A master fingerprint is a noise-free pattern that encodes the unique and immutable characteristics of a “synthetic finger”, independently of the variations (displacement, rotation, pressure, skin condition, distortion, noise, etc.) that make the successive acquisitions different from each other (see Fig. 7.1a). Examples of methods belonging to this category can be found in Cappelli et al. (2000, 2002), Araque et al. (2002), Zhao et al. (2012), Imdahl et al. (2015); Sects. 7.2 and 7.3 describe the main techniques proposed for generating a master fingerprint and those for deriving fingerprint impressions, respectively;
- methods that directly generate each synthetic fingerprint starting from a set of input parameters (see Fig. 7.1b). Examples of methods belonging to this category can be found in Cao and Jain (2018), Attia et al. (2019), and Mistry et al. (2020) and will be described in Sect. 7.4.

Figure 7.2 compares a real fingerprint to synthetic patterns generated by some of the above methods. For each of them, Table 7.1 reports, respectively: the models adopted for generating the orientation image and the ridge-line pattern, the ability to generate multiple fingerprints from the same “synthetic finger” and to generate ground-truth data on relevant features, such as minutiae.

The rest of this chapter is organized as follows. Section 7.5 describes various experiments performed to validate synthetic fingerprints; Sect. 7.6 describes a software tool

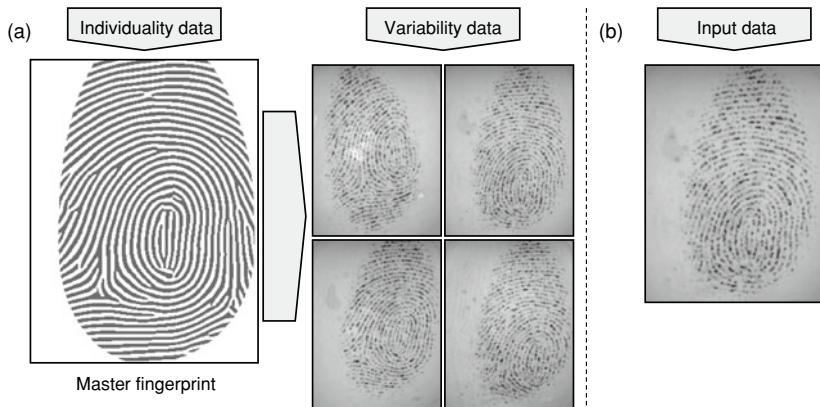


Fig. 7.1 **a** Starting from a set of parameters that represent the unique and immutable features of a “synthetic finger” (individuality data), a master fingerprint is created; then multiple synthetic fingerprints can be generated by changing several parameters that control the appearance (variability data). **b** Starting from a set of input parameters, a synthetic fingerprint is directly generated

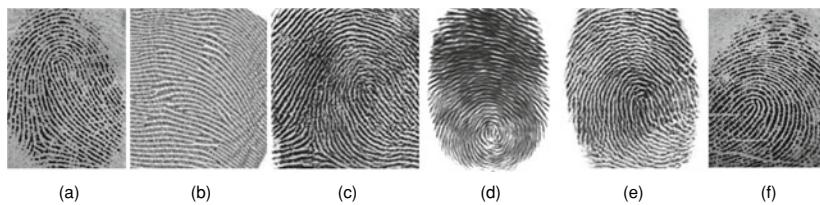


Fig. 7.2 Comparison of synthetic and real fingerprints: **a–e** are generated, respectively, with methods in **a** Cappelli et al. (2000, 2002), **b** Zhao et al. (2012), **c** Cao and Jain (2018), **d** Attia et al. (2019), **e** Mistry et al. (2020); **f** is a real fingerprint

that implements the synthesis technique proposed by Cappelli et al. (2000, 2002). Finally, Sect. 7.7 provides some concluding remarks.

7.2 Generation of a Master Fingerprint

A typical fingerprint recognition algorithm processes a fingerprint as summarized in Fig. 7.3: (i) the fingerprint is first segmented from the background, (ii) the local frequency and orientation maps are estimated, and (iii) this information is exploited to enhance the ridge pattern and find the minutiae. In order to generate a master fingerprint, some of the above operations are usually “inverted” (see Fig. 7.4): a fingerprint area, an orientation

Table 7.1 Fingerprint synthesis methods in the literature. For each of them, the models adopted for generating the orientation image and the ridge-line pattern are reported; the last two columns indicate whether the method can generate multiple fingerprints for the same “finger” and can produce ground-truth data (“Possible” means that the method in principle allows it, but the authors did not explore such a possibility)

Method	Orientation model	Ridge generation	Multiple impressions	Ground-truth
Cappelli et al. (2000, 2002)	Zero-pole	Iterative Gabor	Yes	Yes
Araque et al. (2002)	Zero-pole	Iterative Gabor	Possible	Possible
Zhao et al. (2012)	Zero-pole	AM-FM	Yes	Possible
Imdahl et al. (2015)	Real fingerprint	Iterative Gabor	Possible	Possible
Cao and Jain (2018)	IWGAN/autoencoder		No	No
Attia et al. (2019)	Variational autoencoder		No	No
Mistry et al. (2020)	IWGAN/autoencoder/identity loss		No	No

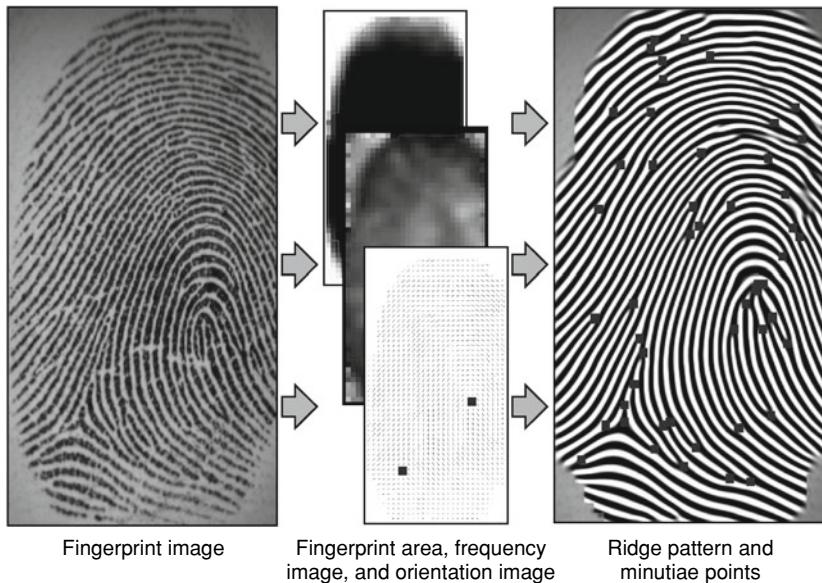
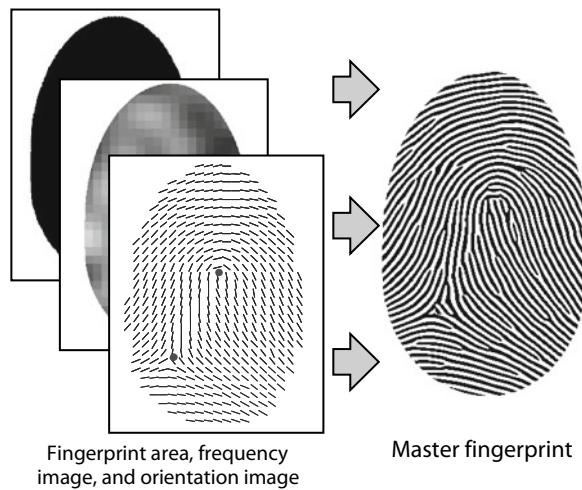


Fig. 7.3 A typical fingerprint feature extraction process

Fig. 7.4 The basic idea behind a typical method for master-fingerprint generation



image, and a frequency image, generated independently of each other, are the inputs to a ridge generation process.

Most of the methods proposed in the literature to create master fingerprints adopt the following steps:

1. Fingerprint area generation,
2. Orientation image generation,
3. Frequency image generation,
4. Ridge pattern generation.

Step 1 defines the external silhouette of the fingerprint; step 2 generates a consistent orientation image, and step 3 creates a frequency image. In Step 4, the ridge line pattern and the minutiae are created to produce a near-binary fingerprint image: the master fingerprint. A separate section is dedicated to each of the above steps.

7.2.1 Fingerprint Area Generation

Depending on various factors such as the finger size, position, and pressure against the acquisition sensor, the fingerprint images have different sizes and external shapes (Fig. 7.5).

Cappelli et al. (2002), after a visual examination of many fingerprint images, suggested that a simple model, based on four elliptical arcs and a rectangle and controlled by five parameters (see Fig. 7.6), can handle most of the variations present in fingerprint shape.



Fig. 7.5 Examples of fingerprint images with different size and shape

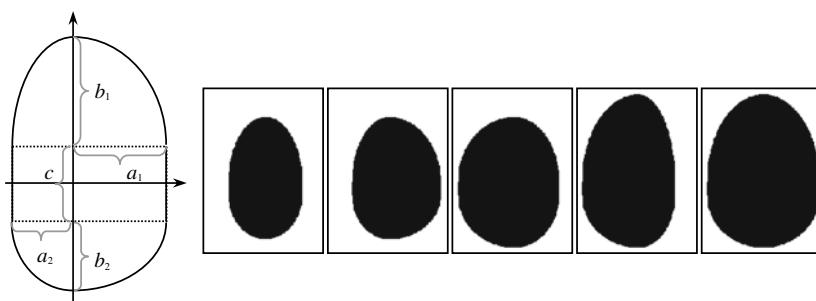


Fig. 7.6 On the left: the fingerprint shape model proposed in Cappelli et al. (2002); on the right: some examples of fingerprint silhouettes generated by the model

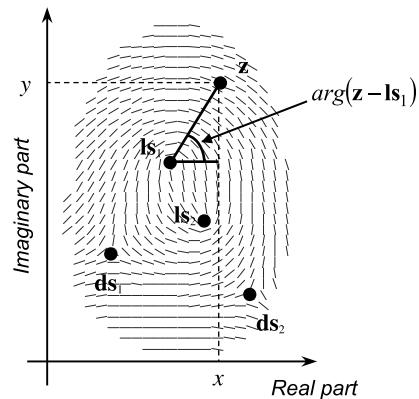
Figure 7.6 shows some examples of fingerprint shapes generated by this model by varying the five parameters. The same model was used by Imdahl et al. (2015).

7.2.2 Orientation Image Generation

The orientation image (see Sect. 3.3) defines the high-level structure of a fingerprint: in fact, it completely defines level-1 features, including the position of the singularities and the fingerprint class. Generating a realistic orientation image is then a crucial step in fingerprint synthesis. A first possibility is to randomly choose the orientation image of a real fingerprint from a database, as in Imdahl et al. (2015): this approach is simple and ensures the quality of the orientation image, but on the other hand it limits the number of orientation images to the size of the real fingerprint database.

Most techniques proposed to synthetize orientation images are based on the *zero-pole model*, proposed by Sherlock and Monro (1993): this model allows a consistent orientation image to be computed from the knowledge of the position of fingerprint singularities (loops and deltas) alone. In this model, the image is located in the complex plane and the local ridge orientation is the phase of the square root of a complex rational function whose singularities (poles and zeros) are located at the same place as the fingerprint singularities

Fig. 7.7 Each element of the orientation image is considered as a complex number (Sherlock & Monro, 1993)



(loops and deltas). Let \mathbf{ls}_i , $i = 1 \dots n_c$ and \mathbf{ds}_i , $i = 1 \dots n_d$ be the coordinates of the loops and deltas, respectively. The orientation $q \in [0^\circ, 180^\circ]$ at each point $\mathbf{z} = [x, y]$ is calculated as

$$\theta = \frac{1}{2} \left[\sum_{i=1}^{n_d} \arg(\mathbf{z} - \mathbf{ds}_i) - \sum_{i=1}^{n_c} \arg(\mathbf{z} - \mathbf{ls}_i) \right], \quad (7.1)$$

where the function $\arg(\mathbf{c})$ returns the phase angle of the complex number \mathbf{c} (see Fig. 7.7).

The zero-pole model may be exploited for generating synthetic orientation images as follows. First a fingerprint class is randomly chosen and then the positions of the singularities are randomly selected according to class-specific constraints (for instance, in a left loop, the delta must be on the right side of the loop). Figure 7.8 shows some examples of orientation images generated by this model. Unfortunately, the generation of synthetic orientation images for arch type patterns that do not contain any singularities is not supported by this model, and it must be considered separately. However, this does not pose a serious problem inasmuch as arch orientation image generation is straightforward, and a simple sinusoidal function (whose frequency and amplitude are tuned to control the arch curvature and aspect) can adequately approximate this pattern (Cappelli et al., 2000).

The ridge line flow in a real fingerprint image cannot be completely determined by the singularity type and position. Hence, although the zero-pole model is a good starting point, it is not satisfactory. Figure 7.8e shows a fingerprint image (belonging to the left loop class) and the orientation image generated by the zero-pole model, with the same position of loop and delta. Clear differences exist among the real ridge line orientations and the corresponding elements in the orientation image: in particular, the regions above the loop and between the loop and the delta are not well modeled. For this reason, more complex models are often used. Zhao et al. (2012) decompose the orientation image into singular and residual components, which are approximated by the zero-pole model and the cosine peripheral model (Wang & Hu, 2011): the parameters of the zero-pole model

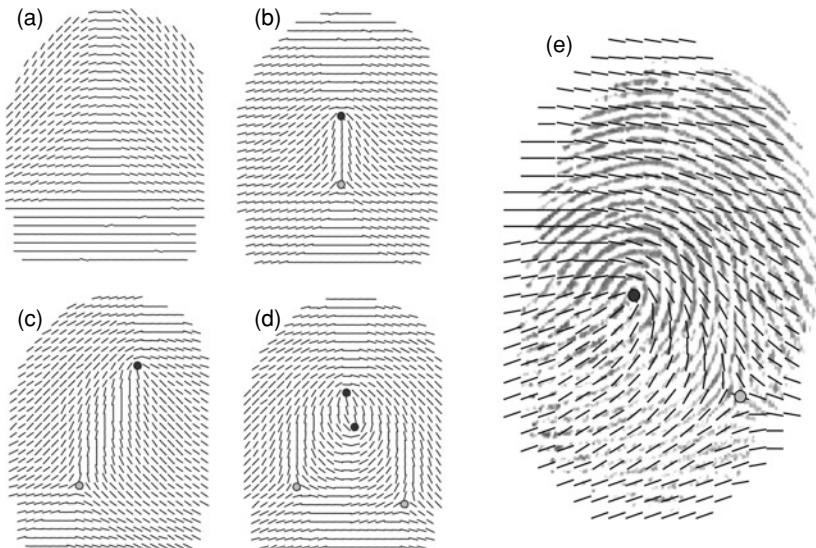


Fig. 7.8 An example of **a** arch, **b** tented arch, **c** right loop, and **d** whorl orientation image as generated by the zero-pole model. **e** An example of a left loop orientation image superimposed over a left loop fingerprint with coincident singularity positions

(positions of the singularities) are sampled according to the statistical models in Cappelli and Maltoni (2009), while the parameters of the cosine peripheral model are sampled from Gaussian mixtures trained over a real fingerprint database.

Araque et al. (2002) and Cappelli et al. (2002) adopt a variant of the zero-pole model proposed by Vizcaya and Gerhardt (1996), which introduces more degrees of freedom to cope with the orientation variability that may characterize orientation images with coincident singularities. In this model the orientation θ at each point \mathbf{z} is calculated as

$$\theta = \frac{1}{2} \left[\sum_{i=1}^{n_d} g_{\mathbf{ds}_i}(\arg(\mathbf{z} - \mathbf{ds}_i)) - \sum_{i=1}^{n_c} g_{\mathbf{ls}_i}(\arg(\mathbf{z} - \mathbf{ls}_i)) \right], \quad (7.2)$$

where $g_k(\alpha)$, for $k \in \{\mathbf{ls}_1, \dots, \mathbf{ls}_{n_c}, \mathbf{ds}_1, \dots, \mathbf{ds}_{n_d}\}$, are piecewise linear functions capable of locally correcting the orientation image with respect to the value given by the zero-pole model:

$$g_k(\alpha) = \bar{g}_k(\alpha_i) + \frac{\alpha - \alpha_i}{2\pi/L} (\bar{g}_k(\alpha_{i+1}) - \bar{g}_k(\alpha_i)), \quad (7.3)$$

for $\alpha_i \leq \alpha \leq \alpha_{i+1}$, $\alpha_i = -\pi + \frac{2\pi i}{L}$.

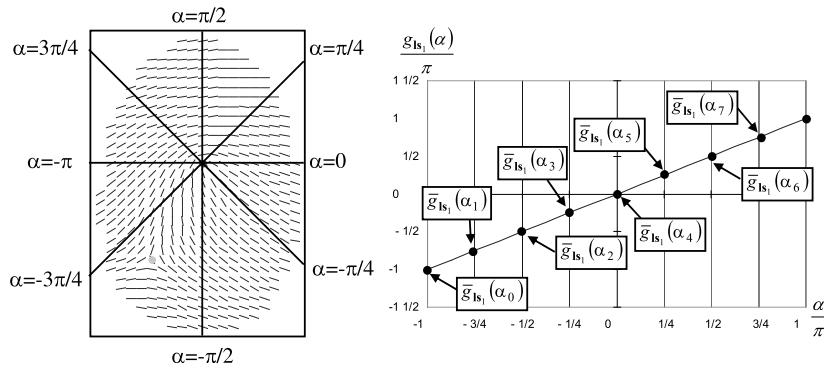


Fig. 7.9 Definition of a function $g_k(\alpha)$ for the loop singularity of a right loop orientation image (Vizcaya & Gerhardt, 1996). In this case, $g_k(\alpha)$ is the identity function and the model coincides with that of Sherlock and Monro (1993)

Each function $g_k(\alpha)$ is defined by the set of values $\{\bar{g}_k(\alpha_i) | i = 0 \dots L - 1\}$, where each value is the amount of correction of the orientation image at a given angle (in a set of L angles uniformly distributed between $-\pi$ and π). If $\bar{g}_k(\alpha_i) = \alpha_i \forall i \in \{0 \dots L - 1\}$ (i.e., $g_k(\alpha)$ is the identity function), the model coincides with the simple zero-pole (see Fig. 7.9).

The aim of the Vizcaya and Gerhardt (1996) work is to approximate a real orientation image, given a specific fingerprint, and so the authors derive the values $\bar{g}_k(\alpha_i)$ through an optimization procedure. In the context of synthetic generation, the Vizcaya and Gerhardt model is not used to approximate the orientation image of a given fingerprint but instead, the additional degrees of freedom are exploited to provide more variations. From the analysis of real fingerprints, Cappelli et al. (2000) found that $L = 8$ is a reasonable value and derived appropriate ranges for the parameters $\bar{g}_k(\alpha_i)$ for each fingerprint class: during the orientation-image generation, random values are selected within such ranges. Actually, in order to produce realistic results, for each singularity k , only $\bar{g}_k(\alpha_0)$ and $\bar{g}_k(\alpha_4)$ are randomly selected: the other values are determined so that a smooth mapping function $g_k(\alpha)$ is obtained. Figure 7.10 shows the effect of changing the parameter $\bar{g}_{ls_1}(\alpha_4)$ in a right loop fingerprint: the changes with respect to the Sherlock and Monro formulation (see Fig. 7.9) are highlighted in the corresponding orientation image.

Figure 7.11a, b show two examples of orientation images generated according to the Vizcaya and Gerhardt model; these images appear to be more realistic than those in Fig. 7.8. The superiority of the Vizcaya and Gerhardt model in approximating real ridge patterns is also evident from the comparison between Figs. 7.11c and d.

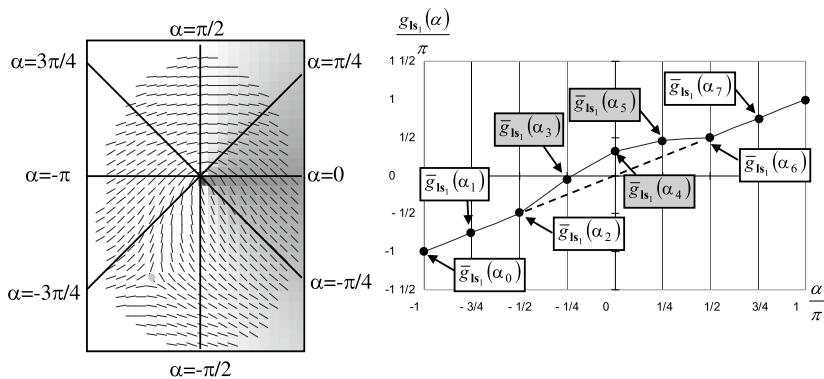


Fig. 7.10 The effects of modifying the parameters that control a mapping function $g_k(\alpha)$ are highlighted in the corresponding orientation image (Vizcaya & Gerhardt, 1996)

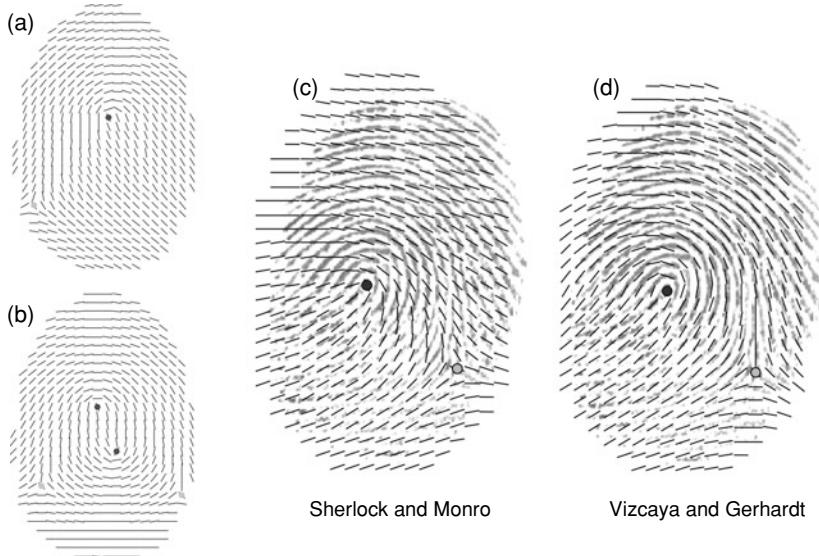
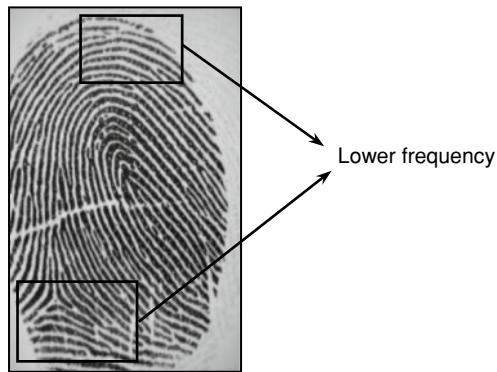


Fig. 7.11 An example of **a** right loop and **b** whorl orientation images, as generated by the Vizcaya and Gerhardt model. In **c** and **d** the orientation images produced by the two models, for a given fingerprint, are compared

Fig. 7.12 An example of a right loop fingerprint, where the ridge line frequency is lower in the regions above the loop and below the delta



7.2.3 Frequency Image Generation

The generation of realistic frequency images is disregarded in most fingerprint synthesis studies: in Araque et al. (2002), Zhao et al. (2012), and Imdahl et al. (2015), the frequency is simply assumed constant over the whole fingerprint pattern.

A more sophisticated approach was proposed in Cappelli et al. (2000), where the authors, from the visual inspection of a large number of fingerprint images, noted that quite often, in the regions above the northernmost loop and below the southernmost delta, the ridge line frequency is lower than in the rest of the fingerprint (see Fig. 7.12). Therefore, their frequency-image generation is performed as follows:

1. A feasible overall frequency is randomly selected according to the distribution of ridge line frequency in real fingerprints; an average ridge/valley period of nine pixels is used. This simulates a 500 dpi sensor.
2. The frequency in the above-described regions is slightly decreased according to the positions of the singularities.
3. The frequency image is randomly perturbed to improve its appearance.
4. A local smoothing by a 3×3 averaging box filter is performed.

Figure 7.13 shows some examples of frequency images synthetically generated by Cappelli et al. (2000).

7.2.4 Ridge Pattern Generation

The ridge-line pattern is the part that most characterizes fingerprints, a kind of pattern that is universally known and associated to the concept of fingerprint individuality; it contains the minutiae, which are the most important features for fingerprint recognition. Studies aimed at modeling the ridge-line pattern (based on some hypothesized physical



Fig. 7.13 Some examples of synthesized frequency images (Cappelli et al., 2000): light blocks denote higher frequencies

mechanisms of fingerprint formation during embryogenesis) precede the first fingerprint synthesis works. In Sherstinsky and Picard (1994), a complex method which employs a dynamic non-linear system called “M-lattice” is introduced with the aim of binarizing a gray-scale fingerprint image. The method is based on the reaction–diffusion model first proposed by Turing (1953) to explain the formation of various patterns observed on animal skin such as zebra stripes. Although Sherstinsky and Picard do not address fingerprint synthesis, the ridge line model they proposed could be used as a basis for synthetic fingerprint generation. Penrose hypothesized that fingerprint patterns such as loops and whorls are formed by ridges corresponding to the lines of curvature of the skin of the embryo at the time when the ridges were being formed (Penrose, 1965). Under this hypothesis, Mardia et al. (1992) demonstrated that fingerprint patterns can be modeled by differential equations having exact solution.

Novikov and Glushchenko (1998) proposed a ridge generation technique operating in the frequency domain. For each pixel $[x, y]$ of an initial random image, the 2D Fourier spectrum of a local window, centered in $[x, y]$, is computed. The highest-energy harmonic (i.e., a pure two-dimensional sinusoid in the spatial domain) is chosen from the Fourier spectrum along the normal to the local ridge orientation at $[x, y]$ (according to an a priori artificially generated orientation image). All the sinusoids are summed, and the result is binarized; the procedure is iteratively repeated until a sufficiently smooth image is obtained. This method has some analogies with the iterative application of Gabor filters in the spatial domain discussed later in this section; in fact, the MTF (Modulation Transfer Function) of a Gabor filter is characterized by two symmetric peaks along the normal to the filter orientation (see Sect. 3.6.2).

Kosz (1999) published some results concerning fingerprint synthesis based on a mathematical model of ridge patterns and minutiae; further details on this technique have been provided online by Bicz (2003). According to this model, a fingerprint can be described as a 2D amplitude and frequency modulated (AM–FM) signal:

$$f(x, y) = \cos(\phi(x, y)), \quad (7.4)$$

where

Fig. 7.14 A simple synthetic pattern generated by Eqs. 7.4–7.6, with $\phi_O(x, y) = 2\pi\sqrt{x^2 + y^2}$ and five minutiae



$$\phi(x, y) = \phi_O(x, y) + \phi_M(x, y) \quad (7.5)$$

is a function that defines the phase of the wave structure as the sum of two parts (ϕ_O , which describes the global “shape” of the ridge lines, and ϕ_M , which describes the minutiae). ϕ_M can simply generate n minutiae by adding n spatially-shifted arctangent functions:

$$\phi_M(x, y) = \sum_{i=1}^n p_i \cdot \arctan\left(\frac{y - y_i}{x - x_i}\right), \quad (7.6)$$

where (x_i, y_i) is the location of minutia i and $p_i \in \{-1, 1\}$ denotes its polarity. Figure 7.14 shows a synthetic pattern generated by using the above equations. Larkin and Fletcher (2007) showed that the above model can be effectively used for fingerprint representation, synthesis, and compression. Starting from their work, Feng and Jain (2011) proposed a novel method to reconstruct fingerprints from minutiae and Zhao et al. (2012) introduced a ridge-line pattern generation approach that retains pre-specified minutiae, which are sampled from the Gaussian mixture based spatial distribution model of minutiae proposed by Chen and Jain (2009).

Cappelli et al. (2000) introduce a generation technique based on Gabor filters; this technique proves to be very simple and at the same time powerful: an initial image is created by randomly placing a few black points into a white image; then, by iteratively enhancing this initial image through Gabor filters (adjusted according to the local ridge orientation and frequency), a consistent and very realistic ridge line pattern gradually appears; in particular, fingerprint minutiae of different types (endings, bifurcations, islands, etc.) are automatically generated at random positions. Gabor filters were introduced in Sect. 3.6.2 as an effective tool for fingerprint enhancement; with respect to Eq. (3.10) in Sect. 3.6.2, Cappelli et al. (2000) use equal values for the standard deviations of the Gaussian envelope along the x - and y -axes:

$$\sigma_x = \sigma_y = \sigma.$$

The filter applied at each pixel has the form:

$$g(x, y : \theta, f) = e^{-((x^2+y^2)/2\sigma^2)} \cdot \cos[2\pi \cdot f \cdot (x \cdot \sin\theta + y \cdot \cos\theta)], \quad (7.7)$$

where θ and f are the corresponding local orientation and frequency, respectively. The parameter σ , which determines the bandwidth of the filter, is adjusted according to the local frequency so that the filter does not contain more than three effective peaks (as in Fig. 3.40). In particular, the value of σ is determined by the solution to the following equation.

$$e^{-\left(\left(\frac{3}{2f}\right)^2 / 2\sigma^2\right)} = 10^{-3}. \quad (7.8)$$

Although one could reasonably expect that iteratively applying “striped” filters to random images would simply produce striped images, very realistic minutiae are generated at random positions. Based on their experiments, Cappelli et al. (2000) argue that minutiae primarily originate from the ridge line disparity produced by local convergence/divergence of the orientation image and by frequency changes. Figures 7.15 and 7.16 show some examples of the iterative ridge line generation process. It can be experimentally found that increasing the number of initial points provides a more irregular ridge pattern richer in minutiae (see Fig. 7.15). This is not surprising because expanding distinct image regions causes interference where regions merge, thus favoring the creation of

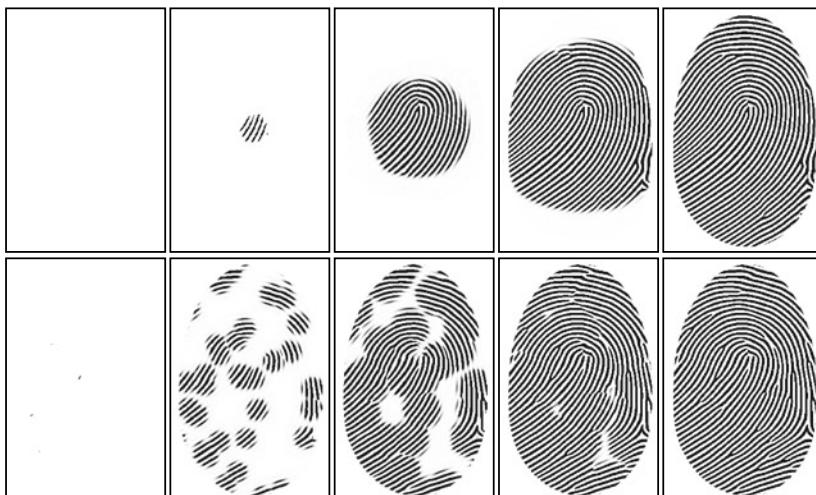


Fig. 7.15 Some intermediate steps of a fingerprint generation process starting from a single central point (top row) and from several randomly located points (bottom row). Usually, increasing the number of initial points provides a more irregular ridge pattern richer in minutiae

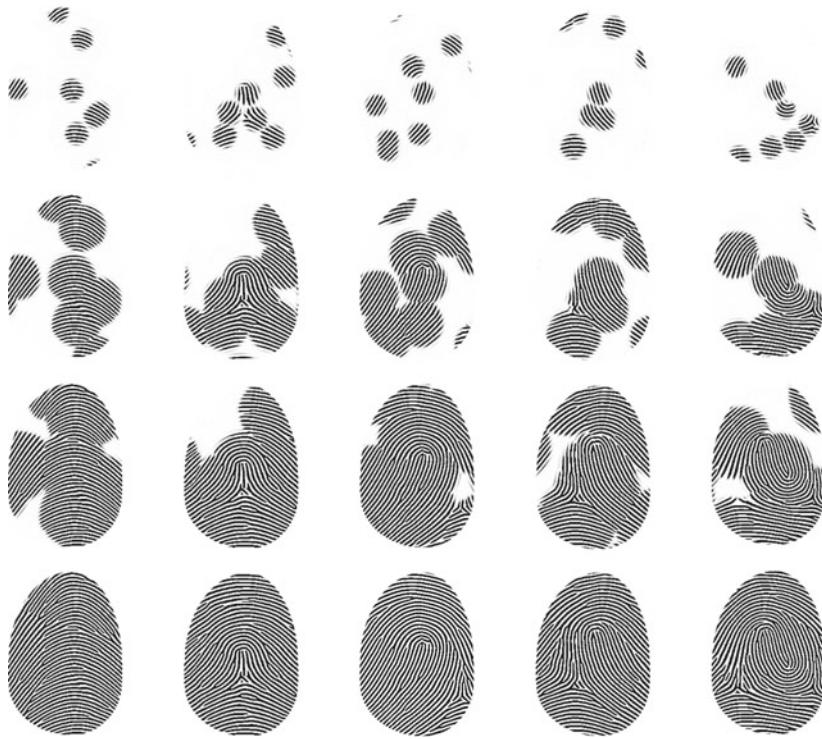


Fig. 7.16 Each column shows an example of a fingerprint generation process for a different fingerprint class; from left to right: arch, tented arch, left loop, right loop, and whorl

minutiae (see Fig. 7.17). A similar technique is used in Araque et al. (2002), where binary directional filters are used instead of Gabor ones, and in Imdahl et al. (2015), where in the initial image each pixel is randomly chosen as black or white, instead of selecting just a few black pixels.

Another ridge-line generation techniques can be found in Hill (2001), where a fingerprint pattern is generated starting from a given set of minutiae points; the work is aimed at proving that a masquerade attack can be carried out against a fingerprint-based biometric



Fig. 7.17 Genesis of a minutia during the fusion of two regions created by two different initial points

system, by “fraudulently” accessing and deciphering the content of a stored template and by recreating an artificial clone (either digital or synthetic). Unfortunately, the algorithm proposed to draw ridge patterns (starting from minutiae positions and then iteratively filling empty spaces) produces images that are visually non-realistic. A more effective ridge reconstruction approach was introduced by Rahmes et al. (2007), with the aim of improving latent fingerprint comparison by automatically restoring low-quality or missing regions in ridge line patterns; however, this approach is only suited to reconstruct a few missing areas in a real fingerprint and cannot be used to create a whole new fingerprint pattern.

Finally, a model for fingerprint formation has been proposed by Kücken and Champod (2013), who assume two influence factors: growth forces, which create mechanical compressive stress on the one hand (see also Kücken & Newell, 2004), and Merkel cells rearranging from a random initial configuration into lines on the other. Computer simulations have shown that an almost periodic pattern very similar to human fingerprints can be generated by applying Kücken’s model: the three main fingerprint classes can be simulated, and minutiae are present in regions where ridge patches with different directions and/or wavelength meet (Kücken, 2007). It is interesting to note that the formation of minutiae according to Kücken’s method has strong analogies with minutiae genesis in the above Gabor-based technique, although the two approaches have been developed starting from totally different hypotheses.

7.3 Generation of Fingerprints from a Master Fingerprint

Several factors contribute to intra-fingerprint variability (making the impressions of a given finger substantially different) when captured by a live-scan scanner (see Chap. 2).

- Displacement in the x - and y -directions and rotation.
- Different finger portions touching the sensor.
- Non-linear distortion produced by the non-orthogonal pressure of the finger against the sensor.
- Variations in the ridge line thickness due to pressure intensity or by skin dampness (wet or dry).
- Small cuts or abrasions on the fingertip.
- Background noise and other sources of noise.

After a master-fingerprint has been created, for each fingerprint impression to be generated from it, some of the following steps are typically performed to simulate the above perturbations.

1. Definition of the fingerprint portion that is in contact with the sensor (this is simply performed by shifting the fingerprint pattern with respect to the fixed external silhouette).
2. Variation in the average thickness of the ridge (Sect. 7.3.1).
3. Simulation of skin distortion (Sect. 7.3.2).
4. Perturbation and rendering (Sect. 7.3.3).
5. Global translation/rotation of the image.
6. Background generation (see Sect. 7.3.4).

7.3.1 Variation in Ridge Thickness

Skin dampness and finger pressure against the sensor platen have similar effects on the acquired images: when the skin is dry or the pressure is low, ridges appear thinner, whereas when the skin is wet or the pressure is high, ridges appear thicker (see Fig. 7.18).

In Cappelli et al. (2002) and in Zhao et al. (2012), morphological operators (Gonzales & Woods, 2007) are applied to the master fingerprint to simulate different degrees of dampness/pressure. In particular, the erosion operator is applied to simulate low pressure or dry skin, and the dilation operator is adopted to simulate high pressure or wet skin (see Fig. 7.19). The structuring element used is a square box whose size varies from 2×2 to 4×4 to modulate the magnitude of the ridge thickness variation.



Fig. 7.18 Three impressions of the same real finger as captured when the finger is dry, normal, and wet, respectively. © IEEE. Reprinted, with permission, from Cappelli et al. (2002)

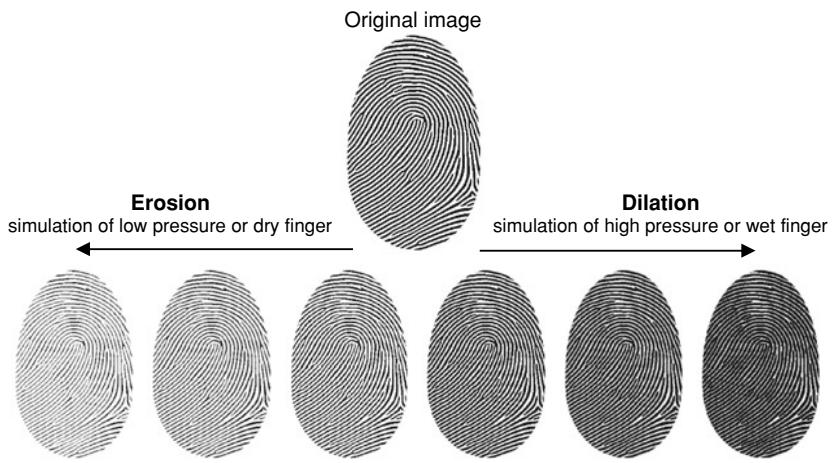


Fig. 7.19 Application of different levels of erosion/dilation to the same master fingerprint. © IEEE. Reprinted, with permission, from Cappelli et al. (2002)

7.3.2 Fingerprint Distortion

One of the main characteristics that distinguishes different impressions of the same finger is the presence of non-linear distortions, mainly due to skin deformations according to different finger placements over the sensing element (see Fig. 7.20). In fact, due to skin plasticity, the application of force, some of whose components are not orthogonal to the sensor surface, produces non-linear distortions (compression or stretching) in the acquired fingerprints (see Chap. 4).



Fig. 7.20 Two impressions of the same real finger where a few corresponding minutiae are marked to highlight the distortion. © IEEE. Reprinted, with permission, from Cappelli et al. (2002)

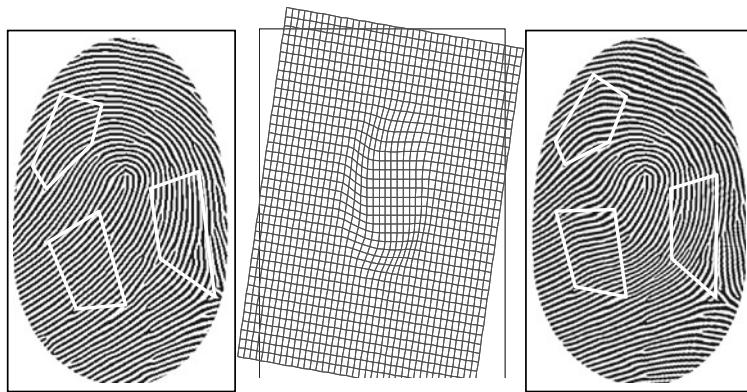


Fig. 7.21 A master fingerprint (on the left) and a distorted impression (on the right); the equivalent distortion of a square mesh is shown in the middle. To better highlight the non-linear deformations, a few corresponding minutiae are connected by white segments in the two images

Cappelli et al. (2002) exploit the skin-distortion model introduced in Cappelli et al. (2001). Unlike in fingerprint matching, where the function `distortion()` (see Sect. 4.5.1) is applied to re-map minutiae points in order to improve fingerprint matching, here the mapping is applied to the whole image, in order to simulate realistic distorted impressions. For this purpose, Lagrangian interpolation is employed to obtain smoothed gray-scale deformed images. Performing Lagrangian interpolation requires the inverse mapping function $\text{distortion}^{-1}()$ to be computed, but unfortunately, this function cannot be analytically expressed. Therefore, for each pixel involved in the mapping, the Newton–Raphson method (Press et al., 1992) is used for numerically calculating the inverse. Figure 7.21 shows a master fingerprint and its distorted impression. A very similar technique is adopted in Zhao et al. (2012). Further approaches to simulate distortion can be found in Cui et al. (2021), where distorted fingerprints are generated for training a convolutional neural network for fingerprint registration, and in Gu et al. (2021), where multiple impressions of the same fingerprint are generated for training CNN-based minutiae descriptors.

7.3.3 Perturbation and Rendering

During fingerprint acquisition, several factors contribute to the deterioration of the true image, thus producing a noisy gray-scale image: irregularity of the ridges and their different contact with the sensor surface, presence of small pores within the ridges, presence of very small prominent ridges, gaps, and clutter noise due to non-uniform pressure of the finger against the sensor. Furthermore, the fingerprint is usually not centered in the image and can present a certain amount of rotation.

Cappelli et al. (2002) sequentially perform the following steps to add perturbations and render the final synthetic impression:

1. Isolate the white pixels associated with the valleys into a separate layer. This is simply performed by copying the pixels brighter than a fixed threshold to a temporary image.
2. Add noise in the form of small white blobs of variable size and shape. The amount of noise increases with the inverse of the fingerprint border distance.
3. Smooth the resulting image with a 3×3 averaging box filter.
4. Superimpose the valley layer to the resulting image.

Steps 1 and 4 are necessary to avoid excessive overall image smoothing. Figure 7.22 shows an example where the intermediate images produced after Steps 2, 4, and 5 are reported. A very similar method is exploited in Zhao et al. (2012).

Cappelli et al. (2004) improved the above technique, by replacing the uniform noise generation in step 3 with more coherent noise (Perlin, 1985), see Fig. 7.23.

Cho et al. (2007) proposed a genetic algorithm-based approach that is able to render new impressions from a given dataset of fingerprints, by selecting a set of filters for simulating different acquisition conditions.

Johnson et al. (2013) proposed a method to model the texture of ridge-line patterns using five features: ridge intensity along the ridge center-lines, ridge width, ridge cross-sectional slope, ridge noise, and valley noise. By modeling these features from a database of real fingerprint images, synthetic images statistically representative of a specific real fingerprint database can be rendered from the master-fingerprints.

7.3.4 Background Generation

In Cappelli et al. (2004) a statistical model based on the KL transform (Jolliffe, 1986) is adopted to generate backgrounds similar to those of fingerprint images acquired with a given sensor. The model requires a set of background-only images as a training set (see Fig. 7.24): a linear subspace that represents the main variations in the training background images is calculated and then used to randomly generate new backgrounds. Formally, let $B = \{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_m\}$ be a set of $m n$ -dimensional vectors (obtained from the background images by concatenating their rows) and let:

- $\bar{\mathbf{b}} = \frac{1}{m} \sum_{\mathbf{b} \in B} \mathbf{b}$ be their mean vector,
- $\mathbf{C} = \frac{1}{m} \sum_{\mathbf{b} \in B} (\mathbf{b} - \bar{\mathbf{b}})(\mathbf{b} - \bar{\mathbf{b}})^T$ be their covariance matrix,
- $\Phi \in \Re^{n \times n}$ be the orthonormal matrix that diagonalizes \mathbf{C} ; that is, $\Phi^T \mathbf{C} \Phi = \Lambda$,

$$\Lambda = \text{Diag}(\lambda_1, \lambda_2, \dots, \lambda_n), \Phi = [\varphi_1, \varphi_2, \dots, \varphi_n],$$

where λ_i and φ_i , $i = 1 \dots n$ are the eigenvalues and the eigenvectors of \mathbf{C} , respectively.

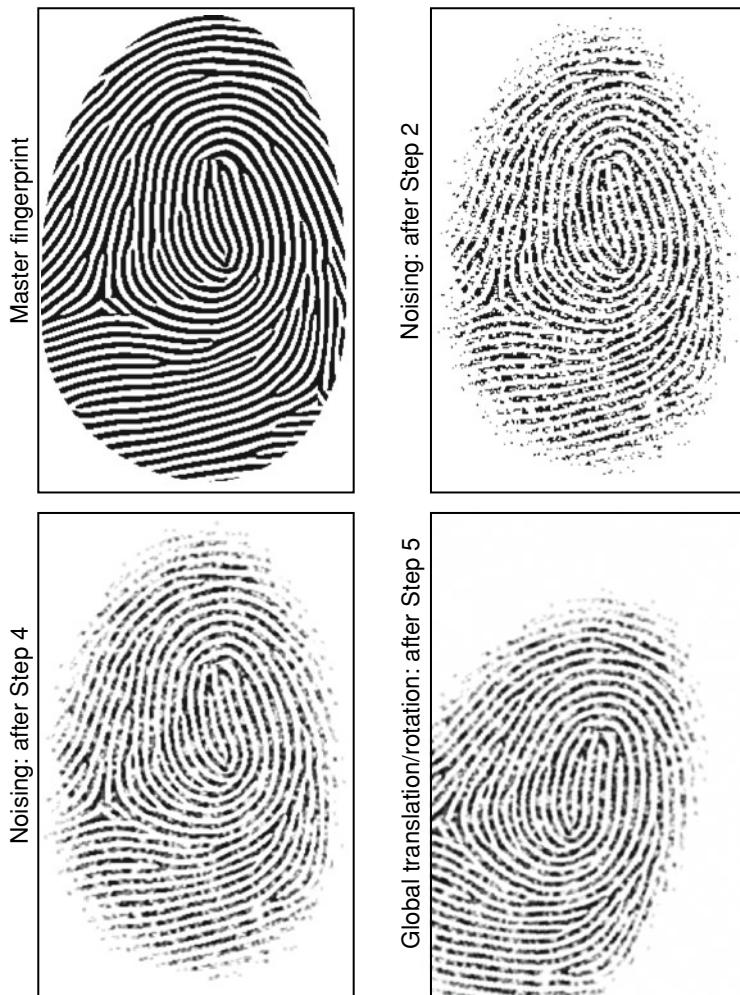


Fig. 7.22 An example of perturbation and global translation/rotation, where the intermediate images produced after Steps 2, 4, and 5 are reported

Then, given a parameter k , $0 < k < \min(n, m)$, the k -dimensional subspace S_B is identified by the mean vector $\bar{\mathbf{b}}$ and by the projection matrix $\Phi_k \in \mathbb{R}^{n \times k}$, whose columns are the k columns of Φ corresponding to the k largest eigenvalues:

$$\Phi_k = [\varphi_{i_1}, \varphi_{i_2}, \dots, \varphi_{i_k}] \quad \text{with} \quad \lambda_{i_1} \geq \lambda_{i_2} \geq \dots \geq \lambda_{i_k} \geq \dots \geq \lambda_{i_n}.$$

The generation of a new background is performed by selecting a point in the subspace S_B and by back projecting it in the original n -dimensional space:



Fig. 7.23 From left to right: an example of uniform noise, an example of Perlin noise, a real fingerprint, a synthetic fingerprint rendered with the approach in Cappelli et al. (2002), and a synthetic fingerprint rendered with the approach in Cappelli et al. (2004)

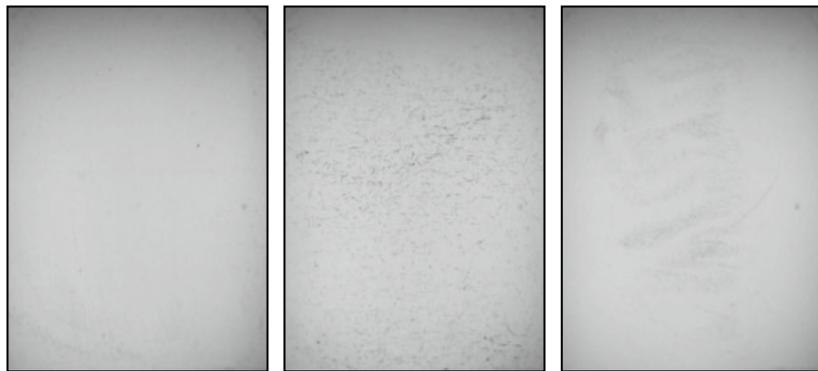


Fig. 7.24 Examples of background-only images (acquired from an optical scanner) used for training the background generator in Cappelli et al. (2004)

1. A k -dimensional vector $\mathbf{y} = [y_1, y_2, \dots, y_k]$ is randomly generated according to k normal distributions: $y_j = N(0, \lambda_{i_j}^{1/2})$, $j = 1 \dots k$
2. The corresponding n -dimensional vector \mathbf{b} is obtained as $\mathbf{b} = \Phi_k \mathbf{y} + \bar{\mathbf{b}}$.

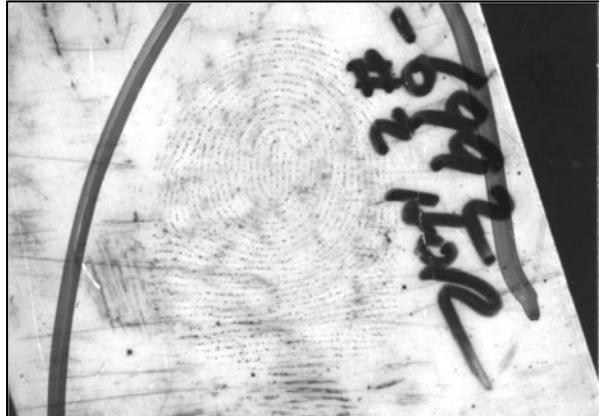
Figure 7.24 shows some examples of the background images (obtained from an optical scanner) used as a training set for the background generation step; Fig. 7.25 reports three synthetic fingerprints with backgrounds generated according to the above model.

The generation of realistic synthetic background is particularly important to simulate latent fingerprints (see Fig. 7.26). A technique to combine partial fingerprint patterns with backgrounds similar to those of latent fingerprints is shown in Rodriguez et al. (2012). In Qian et al. (2019), latent fingerprint noise is simulated by adding structured noise to real fingerprints, with the aim of training a convolutional neural network for latent fingerprint enhancement.



Fig. 7.25 Three synthetic images with backgrounds generated according to the model proposed in Cappelli et al. (2004): the parameters used for training are $m = 65$ and $k = 8$

Fig. 7.26 A synthetic image obtained by combining a noisy fingerprint portion generated by SFinGe (see Sect. 7.6) with a background similar to those of latent fingerprints



7.4 Direct Generation of Synthetic Fingerprints

Some fingerprint synthesis methods do not model the distinctive characteristics of a “synthetic finger” as a master-fingerprint from which to derive the final synthetic impressions; they adopt unsupervised machine learning techniques to train generative models to directly map random vectors into noisy fingerprint images. *Deep generative models*, a combination of generative models and deep neural networks, have shown great promise in image synthesis; in particular, *Variational Autoencoders* (VAEs) (Kingma & Welling, 2013) and *Generative Adversarial Networks* (GANs) (Goodfellow, 2016) are currently the two most popular deep generative models, and both have been investigated for fingerprint synthesis.

Minaee and Abdolrashidi (2018) trained a GAN to generate synthetic images according to a prior distribution sampled from a database of real fingerprints. The discriminator

network contains four convolutional layers, with a fully connected layer at the end; the generator model contains five deconvolution layers. Batch-normalization and leaky rectified linear units are used in both networks. A regularization term, based on anisotropic total variation (simply defined as the sum of the horizontal and vertical components of the gradient of each pixel), is added to the loss function: this improves the ability of the model in learning how to generate ridgelines with a sufficient degree of connectivity.

In Attia et al. (2019), a VAE is trained to generate synthetic fingerprints from latent vectors of 32 elements that follow a normal distribution. The encoder is a network with six convolutional layers that transform the input image into 2048 feature maps of size 4×4 , followed by two other convolution layers that finally produce the two vectors describing the normal distribution of the latent vectors (the former containing the 32 means and the latter the 32 standard deviations). The decoder consists of seven deconvolutional layers that generate the image corresponding to a given latent vector. In an attempt to improve the visual realism of the synthetic fingerprints, the authors added a term to improve the smoothness of the image to the typical loss function adopted in VAEs.

Cao and Jain (2018) experimented I-WGAN, an improved version of the Wasserstein GAN (Gulrajani et al., 2017), initialized by a convolutional autoencoder. Both the autoencoder and the I-WGAN were trained on a database of 250 thousand rolled fingerprints. The generator network has one project-and-reshape layer, followed by seven deconvolutional layers: the first layer generates 4×4 feature maps from a 512-dimensional input random vector; each one of the following layers has a kernel size of 4×4 and a stride size of 2×2 , to successively enlarge the feature map size by a factor of 2, leading to a final output image of size 512×512 . The discriminator network is basically the inverse of the generator. Batch-normalization and leaky rectified linear units are used in both networks, except for the output, which uses *tanh*.

Mistry et al. (2020) noted that while some of the above techniques can generate realistic synthetic fingerprints, especially those trained on large datasets (see Table 7.2), they do not consider the individuality of the generated fingerprints and may continually generate images corresponding to only a few “synthetic fingers”. Starting from the work in Cao and Jain (2018), to encourage the generation of fingerprints which are of distinct identities, they incorporate an additional loss function (named Identity Loss in Fig. 7.27): this loss function is based on the DeepPrint network (see Sect. 4.6.5) and is aimed at minimizing the similarity between the generated fingerprints.

Table 7.2 reports, for each of the above methods, the number of training fingerprints, the size of the input vector, the number of deconvolutional layers in the generator network and the final image size, respectively. Some examples of fingerprint generated by these approaches can be found in Fig. 7.2. A clear advantage of these techniques is that they can be trained to generate images with specific characteristics, by simply providing an appropriate set of real images; for example, rolled or flat fingerprints can be simulated without having to explicitly model their features (Fig. 7.28); on the other hand they suffer from

Table 7.2 A summary of methods that directly generate synthetic fingerprints using deep generative models

Method	Model	Training fingerprints	Input size	Deconvolutional layers (generation network)	Output size
Minaee and Abdolrashidi (2018)	GAN	1,680	100	5	512×512
Cao and Jain (2018)	I-WGAN	250,000	512	7	512×512
Attia et al. (2019)	Variational Autoencoder	800	32	7	512×512
Mistry et al. (2020)	I-WGAN + Identity Loss	280,000	512	7	512×512

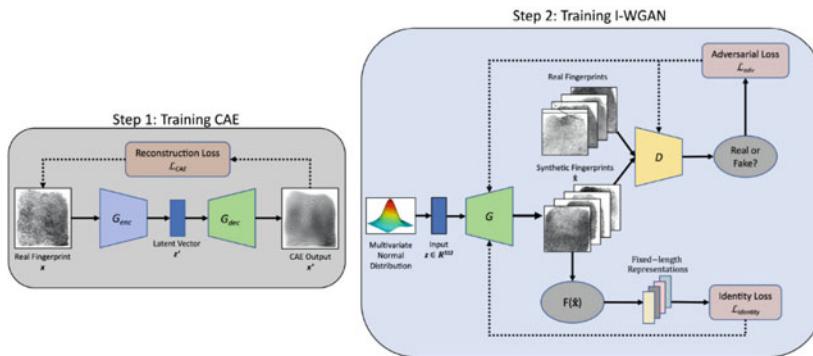


Fig. 7.27 Schema of the fingerprint synthesis approach proposed by Mistry et al. (2020). Step 1 shows the training flow of the convolutional autoencoder (as in Cao & Jain, 2018); step 2 illustrates the training of the I-WGAN (G is the generator, whose weights are initialized using the trained weights of G_{enc} , D is the discriminator) along with the new identity loss function. Solid black arrows show the forward pass of the networks, while dotted black lines show the propagation of the losses. © IEEE. Reprinted, with permission, from Mistry et al. (2020)

a significant limitation: they are not able to generate more fingerprints of the same “synthetic finger”, and this prevents their use in some applications, such as the performance evaluation of fingerprint recognition algorithms requiring intra-class comparisons.



Fig. 7.28 Examples of fingerprint images synthetized by Mistry et al. (2020). The top row shows synthetic rolled fingerprints, while the bottom row plain fingerprints. © IEEE. Reprinted, with permission, from Mistry et al. (2020)

7.5 Validation of Synthetic Generators

Since the early works on synthetic fingerprint generation, researchers have tried to assess the quality of synthetic patterns. Being a new field of study, a standard methodology for validating synthetic fingerprint generators was not available. One of the first attempts in this direction was an experiment carried out during the Fifteenth International Conference on Pattern Recognition (September 2000), when about 90 participants, most of them with some background in fingerprint analysis, were asked to find a synthetic fingerprint image when presented with four images (three of which were real fingerprints and one had been generated by the method proposed in Cappelli et al., 2000). Only 23% of subjects could correctly identify the synthetic image, suggesting that the synthetic pattern appeared visually similar to the real ones. Since then, several techniques have been proposed to validate synthetic fingerprint generators: the following sections describe the most common ones.

7.5.1 Ranking Difference Among Comparison Algorithms

Extensive tests were performed in conjunction with the second International Fingerprint Verification Competition FVC2002 (Maio et al., 2002b). In that contest, four fingerprint databases were used: three of them (DB1, DB2, and DB3) were acquired from real fingers, while fingerprints in the fourth database (DB4) were synthetically generated using the method proposed in Cappelli et al. (2002). An analysis of the ranking distributions among all the participating algorithms over the four FVC2002 databases was performed to understand whether or not the participating algorithms exhibit similar performance on DB4 as on the other databases. Let $R_{ik}^{(j)}$ be the ranking of algorithm i over database k according to the performance indicator j (in FVC2002, the number of participants was 31 and four accuracy indicators were used to compare their performance: EER, ZeroFMR, FMR1000, and FMR100; see Sect. 4.7.2); let $RRD_i^{(j)}$ and $SRD_i^{(j)}$ be the average ranking difference of participant i according to indicator j , among the three real databases and between the synthetic database and each of the real ones, respectively:

$$RRD_i^{(j)} = \frac{\left| R_{i1}^{(j)} - R_{i2}^{(j)} \right| + \left| R_{i1}^{(j)} - R_{i3}^{(j)} \right| + \left| R_{i2}^{(j)} - R_{i3}^{(j)} \right|}{3}, \quad (7.9)$$

$$SRD_i^{(j)} = \frac{\left| R_{i4}^{(j)} - R_{i1}^{(j)} \right| + \left| R_{i4}^{(j)} - R_{i2}^{(j)} \right| + \left| R_{i4}^{(j)} - R_{i3}^{(j)} \right|}{3}. \quad (7.10)$$

$RRD_i^{(j)}$ indicates how stable is the performance of participant i (according to indicator j) over the three databases; $SRD_i^{(j)}$ denotes the amount of variation between synthetic and real databases. Table 7.3 reports, for each indicator $j = 1\dots 4$, a summary of the distribution of $RRD_i^{(j)}$ and $SRD_i^{(j)}$ for $i = 1\dots 31$; the results are somewhat unexpected: the ranking difference $SRD_i^{(j)}$ is often even lower than the corresponding $RRD_i^{(j)}$, indicating that the difference between the synthetic database and the real databases is even smaller than the inter-difference among the three real databases; this shows that a database

Table 7.3 Distributions of $RRD_i^{(j)}$ and $SRD_i^{(j)}$ over all the FVC2002 participating algorithms: average, maximum, minimum values, and the standard deviations are reported for each indicator j

	$RRD_i^{(1)}$	$SRD_i^{(1)}$	$RRD_i^{(2)}$	$SRD_i^{(2)}$	$RRD_i^{(3)}$	$SRD_i^{(3)}$	$RRD_i^{(4)}$	$SRD_i^{(4)}$
Average	2.84	2.65	3.14	2.74	2.58	2.58	2.69	2.59
Max	8.67	11.33	11.33	7.67	7.33	5.67	8.00	10.67
Min	0.00	0.00	0.67	0.33	0.00	0.33	0.00	0.33
St. Dev	2.51	2.43	2.35	1.76	1.94	1.45	2.15	2.36

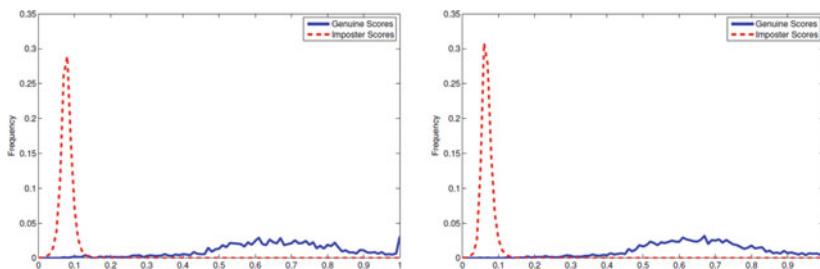


Fig. 7.29 Genuine and imposter match score distributions (Zhao et al., 2012): on a synthetic database (on the left) and a real database (on the right). © IEEE. Reprinted, with permission, from Zhao et al. (2012)

of synthetic fingerprints generated by Cappelli et al. (2002) can be successfully used to measure the performance of matching algorithms.

7.5.2 Match Score Distributions

Zhao et al. (2012) compared the genuine and imposter score distributions obtained from a database of synthetic fingerprints (1000 master-fingerprints, each with five impressions) with those from a real fingerprint database (NIST DB4). The first impression of each finger was used as a reference, and the remaining ones were used as query. Figure 7.29 shows the resulting genuine and imposter score distributions using a fingerprint comparison algorithm. It is worth noting that the distributions for the synthesized fingerprint database have a very similar trend to those for the real fingerprint database. Figure 7.30 shows an analogous comparison for imposter score distributions on synthetic databases created by two deep generative methods: Cao and Jain (2018) and Mistry et al. (2020). Note that genuine score distributions cannot be obtained for these techniques because they are unable to generate multiple impressions of the same “synthetic finger”.

7.5.3 Fingerprint Quality Measures

Performance of fingerprint recognition heavily depends on fingerprint quality (see Sect. 3.11). It is then desirable that the distribution of fingerprint quality over synthetic databases is similar to that on real databases. Cappelli et al. (2018) studied how to improve the fingerprint synthesis technique proposed in Cappelli et al. (2002) to better mimic a given real fingerprint database: one of the metrics they considered was fingerprint quality measured with NFIQ (Sect. 3.11.3); see Fig. 7.31. Other validation experiments based on

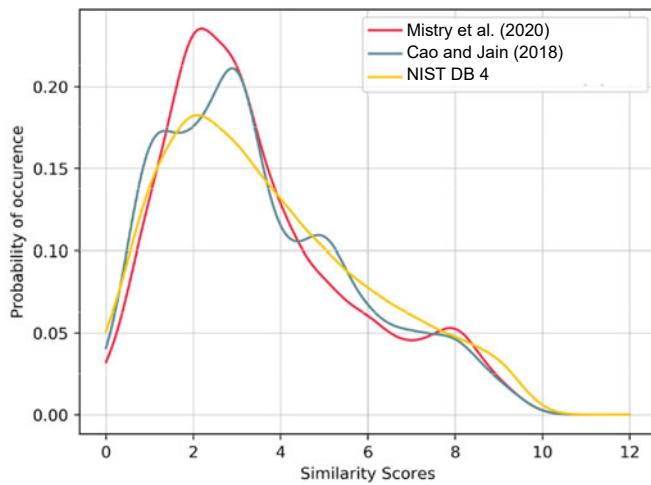


Fig. 7.30 Imposter match score distributions on a synthetic database generated by Mistry et al. (2020), a synthetic database generated by Cao and Jain (2018), and a real database. © IEEE. Reprinted, with permission, from Mistry et al. (2020)

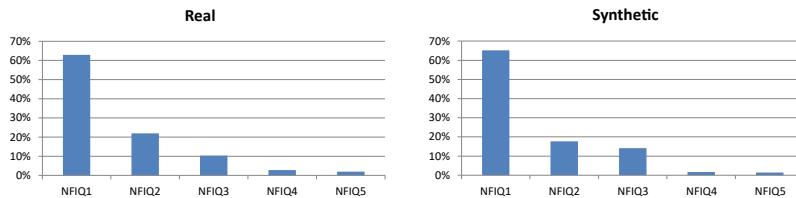


Fig. 7.31 Distribution (histogram) of NFIQ classes: on the left from a real database, on the right from a synthetic database generated by the improved technique described in Cappelli et al. (2018)

fingerprint image quality can be found in Cao and Jain (2018), Attia et al. (2019), and Mistry et al. (2020).

7.5.4 Minutiae Histograms

A two-dimensional minutiae histogram is a fixed-length descriptor of the relative distances and directional differences for all minutiae pairs in a fingerprint: both features (pixel-wise distance and directional difference) are binned by equally sized intervals to build the histogram. Gottschlich and Huckemann (2014) showed that minutiae histograms can help to discriminate synthetic fingerprints generated using the method proposed in Cappelli et al. (2002) from real ones. Imdahl et al. (2015) propose to improve the realism of a

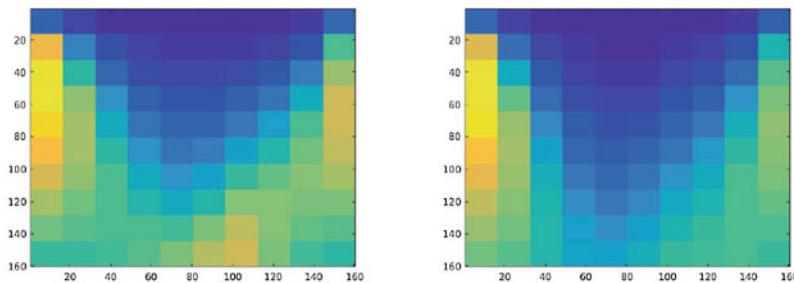


Fig. 7.32 Average 2D minutiae histograms: on the left the histogram was computed over a real database (NIST DB4), on the right over a database generated by the synthesis approach proposed in Cao and Jain (2018). © IEEE. Reprinted, with permission, from Cao and Jain (2018)

synthetic database by generating a larger number of synthetic images and selecting the ones whose minutiae histogram is closer to the average minutiae histogram of a real fingerprint database. Cao and Jain (2018) compare average minutiae histograms of real and synthetic fingerprint databases to validate a new deep generative method (see Fig. 7.32).

7.5.5 Analysis of Multiple Features

Cappelli et al. (2018) used ten features to compare synthetic and real fingerprint databases (Fig. 7.33):

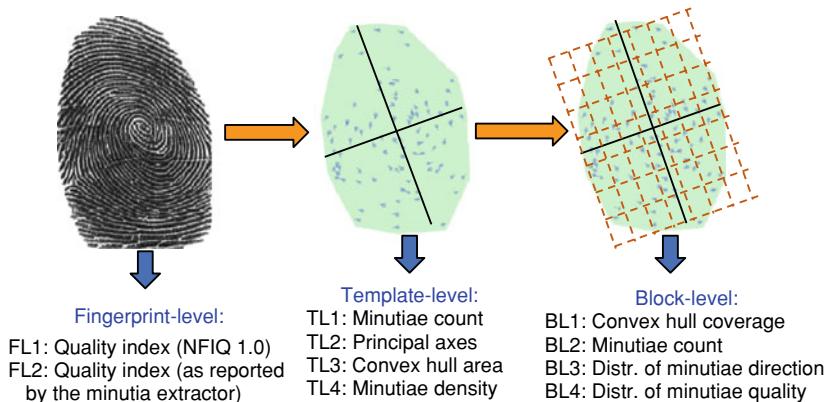


Fig. 7.33 The ten metrics in Cappelli et al. (2018). © IET. Reprinted, with permission, from Cappelli et al. (2018)

- two fingerprint-level quality indices—NFIQ (as already discussed in Sect. 7.5.3), and an algorithm-specific quality index;
- four template-level metrics—number of minutiae, length of the axes of an ellipse fitting minutiae positions, area of the minutiae convex-hull, and density of the minutiae;
- four block-level metrics (for each 16×16 pixel blocks of a grid aligned according to the ellipse axes)—convex-hull coverage, minutiae count, distribution of minutiae directions, and distribution of minutiae quality.

The distributions of the above features, estimated on six real databases, helped Cappelli et al. (2018) to improve the realism of the fingerprint synthesis technique proposed in Cappelli et al. (2002). In the improved method, for each feature, the average difference in the distribution between synthetic and real databases was always lower than or equal to that between pairs of real databases.

Mistry et al. (2020) adopted a similar approach to successfully assess the realism of their new deep generative method with four template-level and three block-level features.

7.5.6 Large Scale Experiments

Running fingerprint recognition experiments on large database (millions of fingerprints) requires addressing two challenges: i) develop a computing platform (synthetic generation software on appropriate hardware resources) to create such a large number of synthetic images in a reasonable time, and ii) use a recognition algorithm fast enough to deal with that large number of fingerprints on the hardware available.

Cappelli et al. (2018) generated a database containing 20 million synthetic fingerprints (ten million “synthetic fingers”, each with two impressions) using an improved and optimized implementation of the method initially proposed in Cappelli et al. (2002), see the following Sect. 7.6 for a description of that software. The database generation took about three days using a cluster of low-cost workstations with a total of 180 CPU cores. Fingerprint verification and identification experiments were performed on that database thanks to an optimized GPU implementation of the MCC algorithm (see Sect. 4.4.3) with a throughput of more than 42 million fingerprint comparisons per second on a single computer. Figure 7.34 reports the results of verification (1:1 comparison) experiments on the synthetic database in terms of five performance indicators: EER, FMR100 (the value of FNMR when FMR = 1%), FMR1000 (the value of FNMR when FMR = 0.1%), FMR10000 (the value of FNMR when FMR = 0.01%), and ZeroFMR (the value of FNMR when FMR = 0%). The constant trend of the performance indicators, except ZeroFMR, is well evident and suggests that the synthesis method has enough “degrees of freedom” to create large databases (there are no “collisions” of synthetic identities due to limitations in the synthetic models). Figure 7.35 reports the results of fingerprint identification (1:N comparison) experiments; three performance indicators are reported:

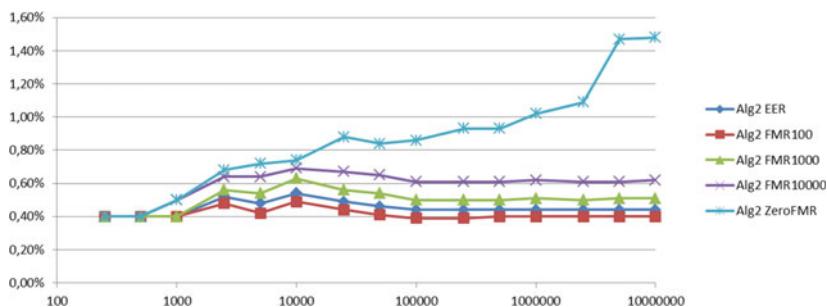


Fig. 7.34 Fingerprint verification results on a synthetic database generated by Cappelli et al. (2018), as a function of the number of fingers in the database (note that the horizontal axis uses a logarithmic scale). © IET. Reprinted, with permission, from Cappelli et al. (2018)

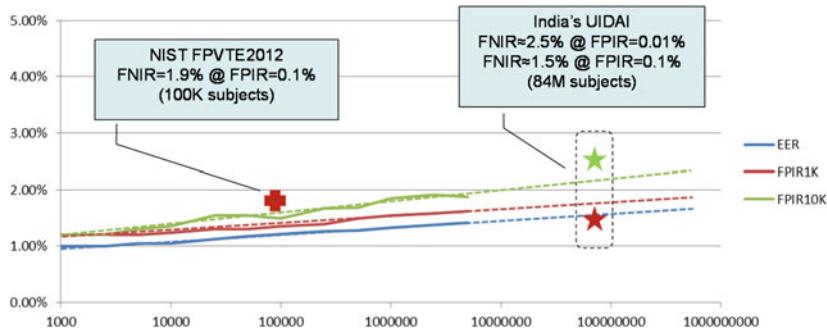


Fig. 7.35 Fingerprint identification results on a synthetic database generated by Cappelli et al. (2018), as a function of the number of fingers in the database (note that the horizontal axis uses a logarithmic scale). Three results from recent large-scale fingerprint identification evaluations on real data have been plot on the graph. The trend of the three indicators (dashed lines) is reported to further predict identification performance increasing the DB size. © IET. Reprinted, with permission, from Cappelli et al. (2018)

EER, FPIR1K (the value of FNIR when FPIR = 0.1%), and FPIR10K (the value of FNIR when FPIR = 0.01%). From the graph, it can be noted that the error rates have a smooth and reasonable trend. The points corresponding to the results of two large scale fingerprint identification evaluations on real data are also plotted on the graph: India's UIDAI project¹ and NIST FPVTE2012 (see Sect. 4.7.2): it can be noted that these results are quite in-line with the error rates obtained on synthetic fingerprint data, suggesting the feasibility of predicting fingerprint identification results with synthetic data.

¹ <https://uidai.gov.in/>.

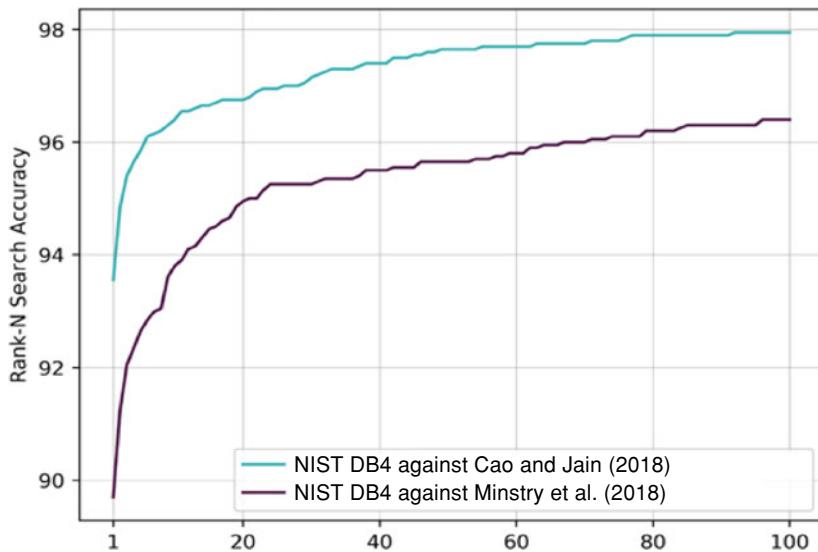


Fig. 7.36 Rank-N search accuracies on NIST DB4 against a synthetic database generated by Cao and Jain (2018) and by Mistry et al. (2020). © IEEE. Reprinted, with permission, from Mistry et al. (2020)

Mistry et al. (2020) generated a database containing 100 million synthetic fingerprints running their deep generative model on a High Performance Computing Center (HPCC) available to them: the generation took 51 CPU hours on computers with 14 cores and a NVIDIA Tesla K80 GPU. Fingerprint identification experiments were carried out by searching NIST DB4 fingerprints against such a large database: the fast fixed-length fingerprint matcher introduced by Engelsma et al. (2021) (see Sect. 4.6.5) was used to compare the fingerprints. This is the largest-scale experiment involving synthetic fingerprints ever reported in the scientific literature. Figure 7.36 shows the rank-N search accuracies on NIST DB4 against a 100 million synthetic database generated by the method proposed in Cao and Jain (2018) and that proposed in Mistry et al. (2020). Noting that the search performance is much lower using fingerprints generated by the latter method, the authors argued that this confirms fingerprints from Mistry et al. (2020) are more unique than those from Cao and Jain (2018).

7.6 The “SFinGe” Software

The Biometric Systems Laboratory of the University of Bologna has developed SFinGe: a software for generating synthetic fingerprints according to the method originally described in Cappelli et al. (2000, 2002). SFinGe (the acronym for Synthetic Fingerprint Generator)

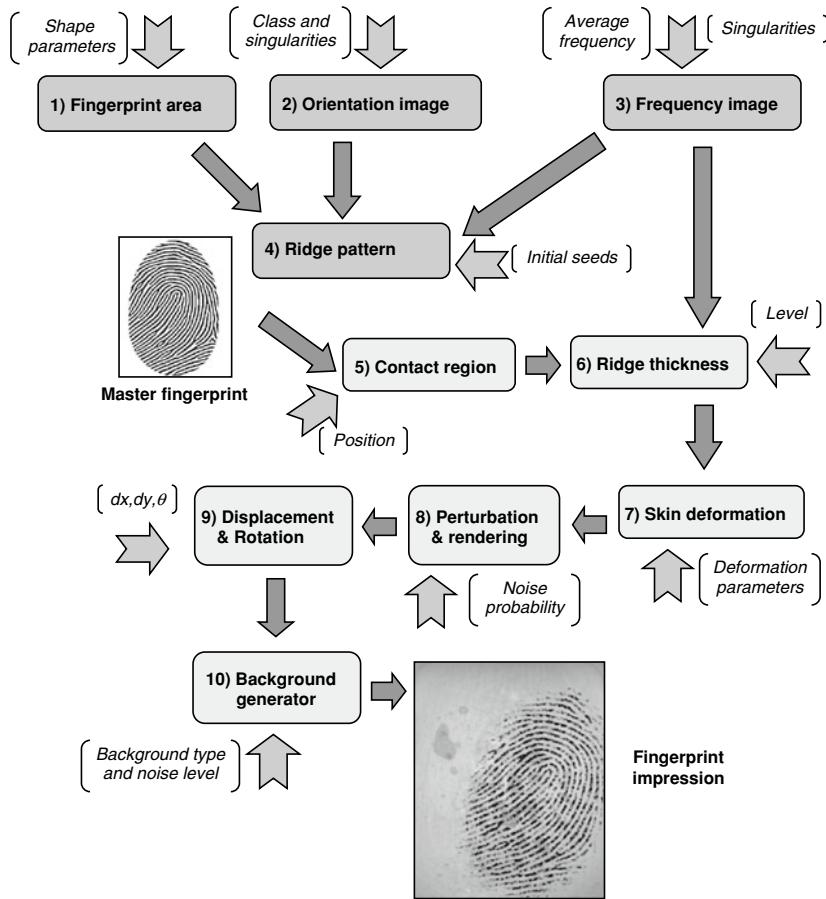


Fig. 7.37 A functional schema of the SFinGe software: each rounded box represents a generation step; the main input parameters that control each step are reported between brackets. Steps 1 to 4 create a master fingerprint; Steps 5 to 10 derive a fingerprint impression from the master fingerprint and can be iterated to produce multiple impressions of the same finger

is the Italian word for “sphinx”. The current version (SFinGe v5.1) includes the improvements described in Cappelli et al. (2004) and in Cappelli et al. (2018). Figure 7.37 shows the fingerprint generation process: Steps 1 to 4 create a master fingerprint; Steps 5 to 10 are performed for each fingerprint derived from the master fingerprint. Figure 7.38 shows some examples of fingerprints generated by SFinGe.

SFinGe is also able to generate the ground truth about relevant features of each synthetic fingerprint, such as the orientation image, the local ridge-line frequency, and the minutiae. The availability of ground-truth is very useful for the development, optimization, and evaluation of feature extraction algorithms, especially for learning-based techniques that usually need large amount of labeled data. Creation of minutiae ground truth is



Fig. 7.38 Two sets of fingerprint impressions (one in each row) generated by SFinGe

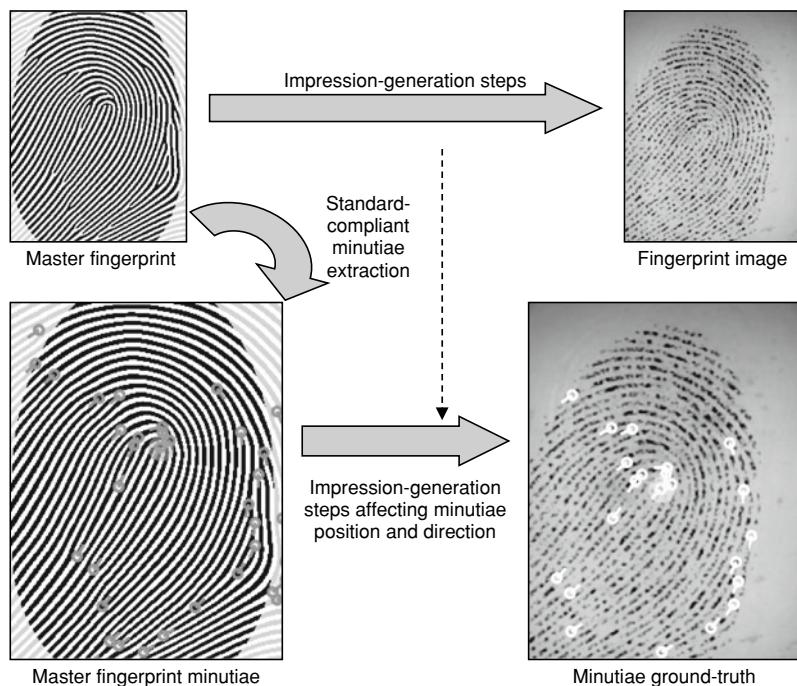


Fig. 7.39 Generation of minutiae ground truth data

performed in parallel with the fingerprint generation (Fig. 7.39). To generate minutiae ground truth, the standard minutiae extraction rules defined in ISO/IEC 19794–2 (2005) are applied to the master fingerprint, then all the relevant transformations executed on the pattern are applied to the minutiae (e.g., translation, rotation, distortion). This approach has some clear advantages:

- The features can be reliably extracted through simple algorithms, since the extraction occurs in a binary image without any noise.
- The ground truth is always unique and accurate, even when the quality of the final image is relatively low (see Fig. 7.40).

Generation of other ground truth features is performed in a similar fashion; for instance, all the relevant transformations are applied to the orientation image calculated at Step 2, thereby obtaining the true orientation image of the final synthetic fingerprint impression.

Figure 7.41 through Fig. 7.43 show the user interface of the software: for each step of the generation method, the user can adjust the main parameters and observe the corresponding effects on the resulting synthetic fingerprint (Fig. 7.43).

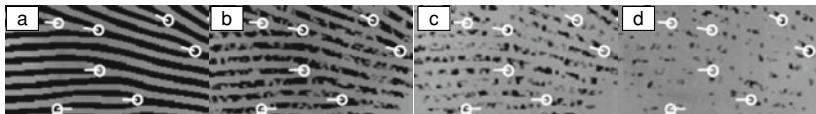


Fig. 7.40 Minutiae ground truth as generated by SFinGe for the same fingerprint portion at different levels of noise: low noise (**a**), medium noise (**b**, **c**), and high noise (**d**)



Fig. 7.41 Main window of the SFinGe software tool

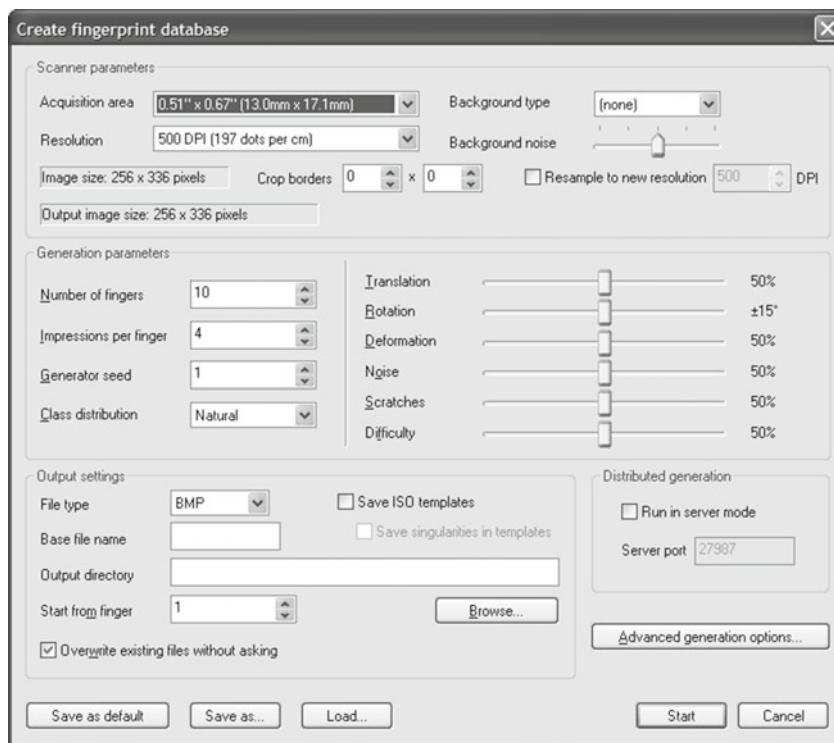


Fig. 7.42 Batch generation options of the SFinGe software

The software allows a database of synthetic fingerprints to be generated in a batch mode, given a relatively small set of input parameters (see Fig. 7.42): number of fingers, impressions per finger, image size and resolution, seed for the random number generator, maximum amount of translation/rotation, maximum amount of perturbation or noise, maximum amount of deformation, and global database difficulty.

The generation of a fingerprint database (including ground-truth data) can be executed in a parallel mode, since each master fingerprint (with its impressions) is independent of the others; this makes it possible to distribute the process on many computers. For instance, using 10 3 GHz PCs in a network, a database of 100,000 fingerprints (10,000 fingers, 10 impressions per finger) can be generated in less than two hours. Thus, a large database of synthetic fingerprints can be generated in a short amount of time. Furthermore, two identical databases can be generated at different places by specifying the same parameters, including the seed for the random number generator: this allows the same test to be reproduced without exchanging huge amounts of fingerprint images.

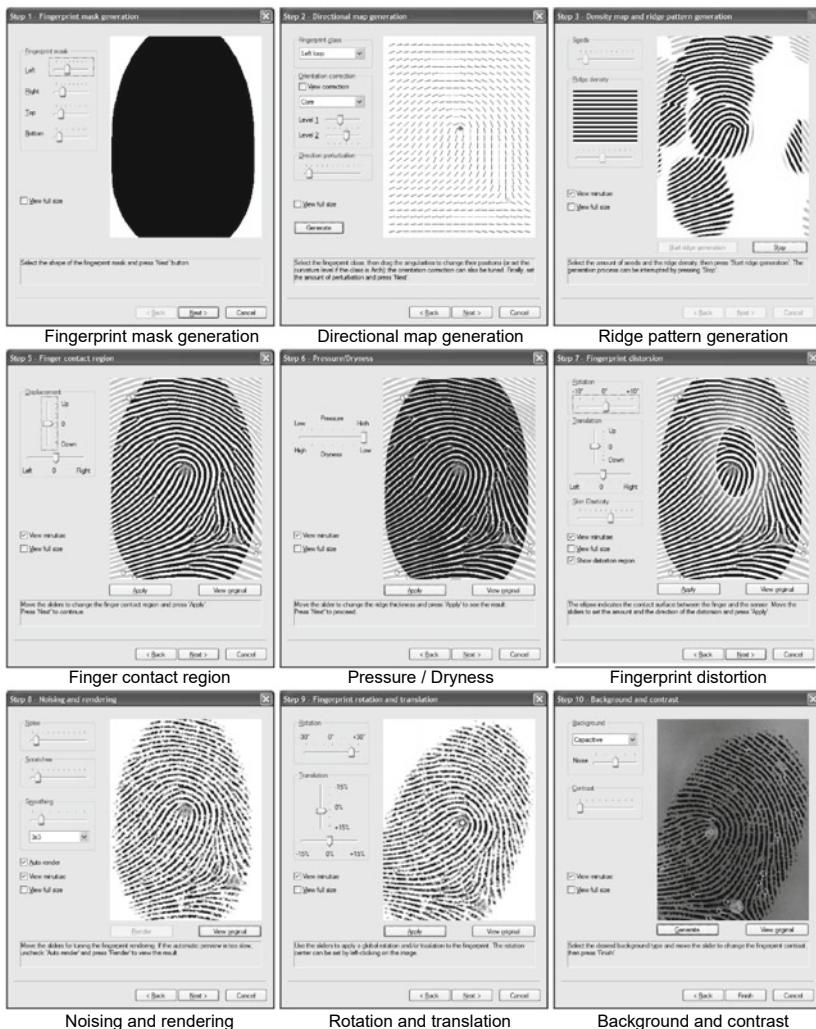


Fig. 7.43 Main steps in fingerprint generation, using the SFinGe software tool

7.7 Summary

Synthetic fingerprint generation is an effective technique to overcome the problem of collecting large fingerprint databases for test purposes. Obviously, synthetic fingerprints are not a substitute for real fingerprint databases, especially when the performance has to be measured with respect to a given application environment and demographics of the user population; on the other hand, synthetic fingerprints have been shown to be well suited for technology evaluations like FVC2000–FVC2006 (Maio et al., 2002a, b, 2004).

Cappelli et al. (2006). The use of synthetic fingerprints is not only limited to the problem of performance evaluation. It can be also used for the following tasks:

- Many machine learning and pattern recognition techniques require a large training set for their learning stage. Synthetic fingerprint images, automatically annotated with ground-truth features, are very well suited for this purpose: in fact, the generator parameters allow explicit control of the type and features of the synthetic fingerprints (e.g., fingerprint type, type of noise, distortion, etc.), and this can be exploited in conjunction with data augmentation and boosting techniques (Schapire, 1990, Freund & Schapire, 1996) to drive the learning process. Supervised deep networks and deep generative models have made use of synthetic fingerprints for training (see Sect. 3.6.4).
- Synthetic fingerprints can be used to test the robustness of fingerprint verification systems against “Trojan horse” attacks at the sensor or the feature extractor module (see Sect. 9.4). Most fingerprint synthesis methods can generate large sets of fingerprints whose features (e.g., minutiae distribution) can be varied independently of other fingerprint characteristics (e.g., orientation image) and, therefore, they are well suited to study the robustness against “hill-climbing” types of Trojan horse attacks and against template reverse engineering, as discussed in Cappelli et al. (2007) and in Feng and Jain (2011).
- Interoperability tests, such as MINEX (Grother et al., 2006) and MTIT (Bazin & Mansfield, 2007), have shown that the location, direction, and type of minutiae extracted by different minutiae extraction algorithms from the same finger image tend to be different. Algorithms *syntactically compliant* to standards such as ISO/IEC 19794-2, (2005), are often not *semantically compliant* (that is, they adopt different rules and conventions to define minutiae placement and direction) and this creates huge interoperability problems. Unfortunately, testing semantic conformance to a minutiae extraction standard is not easy, since it requires a lot of data with manually-labeled minutiae points (ground-truth); furthermore, in low-quality image areas, even manual labeling of minutiae points is not reliable. The automatic generation of ground-truth data for synthetic fingerprint images (see Sect. 7.6) is an effective way to carry out semantic conformance and interoperability studies.

Further investigations are necessary to better understand how similar the synthetic fingerprints are to the real ones from an “individuality” point of view (ref. Chap. 8). Some experimentation in this direction has been performed by analyzing and comparing the distributions of various features (Cappelli et al., 2018, Mistry et al., 2020), suggesting that the most recent methods can properly emulate the variations in real fingerprints. However, a more in-depth analysis is necessary to validate these synthesis techniques.

Finally, it may be worth to investigate new methods that combine the benefits of deep generative models (flexibility and realism) to the advantages of master-fingerprint approaches (ground truth generation and multiple impressions).

References

- Araque, J. L., Baena, M., Chalela, B. E., Navarro, D., & Vizcaya, P. R. (2002). Synthesis of fingerprint images. In *Proceedings of International Conference on Pattern Recognition (16th)* (Vol. 2, pp. 442–445).
- Attia, M., Attia, M. H., Iskander J., Saleh, K., Nahavandi, D., Abobakr, A., Hossny, M., & Nahavandi, S. (2019). Fingerprint synthesis via latent space representation. In *Proceedings of International Conference on Systems, Man and Cybernetics* (pp. 1855–1861).
- Bazin, A. I., & Mansfield, T. (2007). An investigation of minutiae template interoperability. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 13–18).
- Bicz, W. (2003). The idea of description (Reconstruction) of fingerprints with mathematical algorithms and history of the development of this idea at optel. available at: <https://optel.eu/article/polksa/idea.html>. Retrieved November 27, 2008.
- Buettner, D. J., & Orlans, N. M. (2005). A taxonomy for physics based synthetic biometric models. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 10–14).
- Cao, K., & Jain, A. K. (2018). Fingerprint synthesis: Evaluating fingerprint search at scale. In *Proceedings of International Conference on Biometrics*.
- Cappelli, R., Maio, D., & Maltoni, D. (2000). Synthetic fingerprint-image generation. In *Proceedings of International Conference on Pattern Recognition* (Vol. 3, pp. 475–478).
- Cappelli, R., Maio, D., & Maltoni, D. (2001). Modelling plastic distortion in fingerprint images. In *Proceedings of International Conference on Advances in Pattern Recognition* (pp. 369–376).
- Cappelli, R., Maio, D., & Maltoni, D. (2002). Synthetic fingerprint-database generation. In *Proceedings of International Conference on Pattern Recognition*.
- Cappelli, R., Maio, D., & Maltoni, D. (2004). An improved noise model for the generation of synthetic fingerprints. In *Proceedings of International Conference on Control, Automation, Robotics, and Vision*.
- Cappelli, R., Maio, D., Maltoni, D., Wayman, J. L., & Jain, A. K. (2006). Performance evaluation of fingerprint verification systems. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 28(1), pp. 3–18.
- Cappelli, R., Lumini, A., Maio, D., & Maltoni, D. (2007). Fingerprint image reconstruction from standard templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9), 1489–1503.
- Cappelli, R., & Maltoni, D. (2009). On the spatial distribution of fingerprint singularities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(4), 742–448.
- Cappelli, R., Ferrara, M., & Maltoni, D. (2018). Generating synthetic fingerprints. In M. Drahanský (Ed.), *Hand-Based biometrics: Methods and technology*. IET.
- Chen, Y., & Jain, A. K. (2009). Beyond minutiae: A fingerprint individuality model with pattern, ridge and pore features. In *Proceedings of International Conference on Biometrics*.
- Cho, U. K., Hong, J. H., & Cho, S. B. (2007). Automatic fingerprints image generation using evolutionary algorithm. In *Proceedings of International Conference on Biometrics* (pp. 134–143).
- Cui, Z., Feng, J., & Zhou, J. (2021). Dense registration and mosaicking of fingerprints by training an end-to-end network. *IEEE Transactions on Information Forensics and Security*, 16, 627–642.

- Engelsma, J. J., Cao, K., & Jain, A. K. (2021). Learning a fixed-length fingerprint representation. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 43(6), 1981–1997.
- Feng, J., & Jain, A. K. (2011). Fingerprint reconstruction: From minutiae to phase. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(2), 209–223.
- Freund, Y., & Schapire, R. (1996). Experiments with a new boosting algorithm. In *Proceedings of International Conference on Machine Learning* (pp. 148–156).
- Gonzales, R. C., & Woods, R. E. (2007). *Digital image processing* (3rd ed.) Prentice-Hall, Englewood Cliffs.
- Goodfellow, I. (2016). NIPS 2016 tutorial: Generative adversarial networks. [arXiv:1701.00160](https://arxiv.org/abs/1701.00160).
- Gottschlich, C., & Huckemann, S. (2014). Separating the real from the synthetic: Minutiae histograms as fingerprints of fingerprints. *IET Biometrics*, 3(4), 291–301.
- Grother, P., McCabe, M., Watson, C., Indovina, M., Salamon, W., Flanagan, P., Tabassi, E., Newton, E., & Wilson, C. (2006). *Performance and interoperability of the INCITS 378 fingerprint template*. NIST Research Report: NISTIR 7296.
- Gu, S., Feng, J., Lu, J., & Zhou, J. (2021). Latent fingerprint registration via matching densely sampled points. *IEEE Transactions on Information Forensics and Security*, 16, 1231–1244.
- Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. (2017). Improved training of Wasserstein GANs. [arXiv:1704.00028](https://arxiv.org/abs/1704.00028).
- Hill, C. J. (2001). *Risk of masquerade arising from the storage of biometrics*. Bachelor of Science Thesis, The Department of Computer Science Australian National University.
- ISO/IEC 19794-2 (2005). ISO/IEC, “ISO/IEC 19794-2:2005 – Biometric Data Interchange Formats – Part 2: Finger Minutiae Data”. ISO/IEC Standard.
- Imdahl, C., Huckemann, S., & Gottschlich, C. (2015). Towards generating realistic synthetic finger print images. In *Proceedings of International Symposium on Image and Signal Processing and Analysis* (pp. 78–82).
- Johnson, P. A., Hua, F., & Schuckers, S. A. C. (2013). Texture modeling for synthetic fingerprint generation. In *Proceedings of CVPR Workshop on Biometrics* (pp. 154–159).
- Jolliffe, I. T. (1986). *Principle component analysis*. Springer, New York.
- Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. [arXiv:1312.6114](https://arxiv.org/abs/1312.6114).
- Kosz, D. (1999). New numerical methods of fingerprint recognition based on mathematical description of arrangement of dermatoglyphics and creation of minutiae. In D. Mintie (Ed.), *Biometrics in Human Service User Group Newsletter*.
- Kücken, M. (2007). Models for fingerprint pattern formation. *Forensic Science International*, 171(2–3), 85–96.
- Kücken, M., & Newell, A. C. (2004). A model for fingerprint formation. *Europhysics Letters*, 68(1), 141.
- Kücken, M., & Champod, C. (2013). Merkel cells and the individuality of friction ridge skin. *Journal of Theoretical Biology*, 317, 229–237.
- Larkin, K. G., & Fletcher, P. A. (2007). A coherent framework for fingerprint analysis: Are fingerprints holograms? *Optics Express*, 15(14), 8667–8677.
- Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L., & Jain, A. K. (2002a). FVC2000: Fingerprint verification competition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3), 402–412.
- Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L., & Jain, A. K. (2002b). FVC2002: Second fingerprint verification competition. In *Proceedings of International Conference on Pattern Recognition*.
- Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L., & Jain, A. K. (2004). FVC2004: Third fingerprint verification competition. In *Proceedings of International Conference on Biometric Authentication* (pp. 1–7).

- Mardia, K. V., Li, Q., & Hainsworth, T. J. (1992). On the Penrose hypothesis on fingerprint patterns. *IMA Journal of Mathematics Applied in Medicine*, 9(4), 289–294.
- Minaee, S., & Abdolrashidi, A. (2018). Finger-GAN: Generating realistic fingerprint images using connectivity imposed GAN. [arXiv:1812.10482](https://arxiv.org/abs/1812.10482).
- Mistry, V., Engelsma, J. J., & Jain, A. K. (2020). Fingerprint synthesis: Search with 100 million prints. In *Proceedings of International Joint Conference on Biometrics* (pp. 1–10).
- Novikov, S. O., & Glushchenko, G. N. (1998). Fingerprint ridges structure generation models. In *Proceedings of SPIE (International Workshop on Digital Image Processing and Computer Graphics (6th): Applications in Humanities and Natural Sciences)* (Vol. 3346, pp. 270–274).
- Penrose, L. S. (1965). Dermatoglyphic topology. *Nature*, 205, 545–546.
- Perlin, K. (1985). An image synthesizer. *Computer Graphics*, 19(3), 287–296.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (1992). *Numerical Recipes in C*, Cambridge University Press, Cambridge.
- Qian, P., Li, A., & Liu, M. (2019). Latent fingerprint enhancement based on DenseUNet. In *Proceedings of International Conference on Biometrics* (pp. 1–6).
- Rahmes, M., Allen, J. D., Elharti, A., & Tenali, G. B. (2007). Fingerprint reconstruction method using partial differential equation and exemplar-based inpainting methods. In *Proceedings of Biometric Symposium*.
- Rodriguez, C. M., de Jongh, A., & Meuwly, D. (2012). Introducing a semi-automatic method to simulate large numbers of forensic fingermarks for research on fingerprint identification. *Journal of Forensic Sciences*, 57, 334–342.
- Schapire, R. E. (1990). The strength of weak learnability. *Machine Learning*, 5, 197–227.
- Sherlock, B. G., & Monro, D.M. (1993). A model for interpreting fingerprint topology. *Pattern Recognition*, 26(7), 1047–1055.
- Sherstinsky, A., & Picard, R. W. (1994). Restoration and enhancement of fingerprint images using M-Lattice—A novel non-linear dynamical system. In *Proceedings of International Conference on Pattern Recognition*.
- Turing, A. (1953). The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society*, 237, 37–72.
- Vizcaya, P. R., & Gerhardt, L. A. (1996). A nonlinear orientation model for global description of fingerprints. *Pattern Recognition*, 29(7), 1221–1231.
- Wang, Y., & Hu, J. (2011). Global ridge orientation modeling for partial fingerprint identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1), 72–87.
- Yanushkevich, S. N., Stoica, A., Shmerko, V. P., & Popel, D. V. (2005). *Biometric Inverse Problems*, Taylor & Francis/CRC Press.
- Zhao, Q., Jain, A. K., Paultre, N. G., & Taylor, M. (2012). Fingerprint image synthesis based on statistical feature models. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems* (pp. 23–30).



Fingerprint Individuality

8

Abstract

Scientific evidence supporting fingerprint individuality (i.e., quantifying the extent of uniqueness of a fingerprint) is being increasingly demanded, particularly in forensic applications when a suspect is tried for conviction in a court of law. This has generated interest in designing fingerprint individuality models which will quantify the evidential value of fingerprints. This chapter introduces both theoretical and empirical studies on fingerprint individuality: the former is based on statistical models taking into account inter-class and intra-class pattern variations, the latter relying on modeling match (similarity) scores (given a matcher and a corpus of operational fingerprint data). Fingerprint persistence over time is finally addressed in the last section of the chapter.

Keywords

Individuality • Uniqueness • Forensic evidence • Random correspondence • Persistence • Longitudinal study

8.1 Introduction

Expert testimony based on forensic evidence (such as handwriting, fingerprint, hair, bite marks, etc.) is routinely collected at crime scenes and then presented in courtrooms (Houck & Siegel, 2009). Among the various sources of evidence, fingerprints have been

Portions reprinted with permission from *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1010–1025, 2002. © 2002 IEEE.

used in courts of law for almost 100 years. Further, testimony based on fingerprints carries substantial credibility and weight. The use of fingerprint evidence involves comparing salient features of a latent print lifted from a crime scene with those of rolled (full) impressions taken either from the known defendant or those in the criminal and possibly civil fingerprint databases. A reasonably high degree of similarity between the salient features of query latent and database fingerprints (rolled or slap) is the basis for the latent fingerprint experts to testify irrefutably that the person leaving the latent print at the crime scene and the defendant are the same person.

For decades, the testimonies provided by latent experts were almost never excluded from these cases, and on cross-examination, the foundations and basis of such testimonies were rarely questioned (Cole, 2001a, b). Central to establishing an identity based on fingerprint evidence is the assumption of discernible uniqueness; salient features of different individuals are observably different, and therefore, when two prints share many common features, the experts conclude that the owner of the two different prints (latent and its mate in the database) is the same person. The assumption of discernible uniqueness (Saks & Koehler, 2005), although lacking sound theoretical and empirical foundations, allows forensic experts to offer unquestionable proof towards the defendant's guilt, and to make matters worse, these experts are never questioned on the uncertainty associated with their testimonies (that is, how frequently would an observable match between a pair of prints lead to errors in the identification of individuals; see Haber and Haber (2004). Thus, discernible uniqueness precludes the opportunity to establish error rates that would be known from collecting population samples, analyzing the inherent feature variability, and reporting the corresponding probability of two different persons sharing a set of common fingerprint features.

A significant break from this trend occurred in the 1993 case of *Daubert v. Merrell Dow Pharmaceuticals* (113 S. Ct. 2786), where the U.S. Supreme Court ruled that in order for expert forensic testimony to be allowed in a court case, it had to be subject to three main criteria of scientific validation, that is, whether the particular tool or methodology in question (i) has been objectively tested, (ii) has been subjected to peer-review, and (iii) possesses known error rates. Following *Daubert*, fingerprint identification was first challenged in the 1999 case of *U.S. v. Byron Mitchell* (Criminal Action No. 96-407, US District Court for the Eastern District of Pennsylvania) under the premise that the uniqueness of fingerprints has not been objectively tested and matching error rates are unknown (also see Newman, 2001). As a consequence of the outcome of *U.S.v. Byron Mitchell*, fingerprint-based identification has been challenged in several court cases in the United States (for example, *U.S. v. LleraPlaza* [179 F Supp 2d 492 ED Pa 2002 and 188 FSUPP 2d 549 Ed Pa 2002] and *U.S.v. Crisp* [324 F 3d 261 4th Cir 2003]).

Cole (2005) has compiled a list of 22 known exposed cases of erroneous fingerprint identifications made by forensic experts, including the case of Brandon Mayfield (Chap. 6) and the case of Stephan Cowans, who was convicted based on fingerprint evidence and then exonerated by DNA evidence.

In December 2005, the Massachusetts Supreme Judicial Court barred key fingerprint evidence obtained from several latent prints in the case of Terry L. Patterson (Saltzman, 2005a, b). As recently as October 2007, in the case of State of Maryland v. Bryan Rose (Circuit Court, case number K06-0545), judge Susan Souder ruled to exclude fingerprint evidence “*because the State did not prove in this case that opinion testimony by experts regarding the ACE-V (Analysis, Comparison, Evaluation, and Verification) method of latent print identification rests on a reliable factual foundation as required by MD Rule 5-702*”. In making this ruling, the judge heavily relied on the case of Brandon Mayfield. These court rulings demonstrate both the awareness and the need to develop objective measures that reflect the confidence in a match when fingerprints are used as evidence in the courts of law. This problem was also emphasized in a report issued by the US National Research Council (NRC), which reviewed the state of forensic science in the United States (NRC, 2009).

Fingerprint individuality deals with the problem of quantifying the extent of uniqueness of a fingerprint. How similar should two fingerprints be before we can conclude with high confidence that they are from the same finger? What are the measures of fingerprint individuality that reflect the extent of uncertainty in the observed match?

The main challenge in studying fingerprint individuality is to develop statistical models that adequately describe the variability of fingerprint features in a target population. These models can, in turn, be used to derive the probability of a random match between two different fingerprints picked arbitrarily from the target population. Eliciting candidate models for representing the variability of fingerprint features is not an easy task due to the complex nature of this variability. Candidate models should satisfy two important requirements, namely, (i) flexibility, that is, the models can represent a wide range of distributional characteristics of fingerprint features in the population, and (ii) associated confidence measures can be easily obtained from these models.

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 or more 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; (iii) given two fingerprints from two different fingers, determine the probability that they are sufficiently similar (*probability of a random correspondence*). When the comparison is made by an automated system (AFIS), the probability of random correspondence coincides with the false match rate (FMR). Formulation (iii) is more general as its solution would also provide solutions to the other two formulations (Rice, 1995). A reliable statistical estimate of the matching error in fingerprint comparison can determine the admissibility of fingerprint recognition in the courts of law as an evidence of identity. Furthermore, it can establish an upper bound on the performance of automatic fingerprint recognition systems.

In order to solve the individuality problem, one needs to define a priori the representation of the fingerprint. Fingerprints can be represented by several different features, including the overall ridge flow pattern, ridge frequency, number and position of singularities (loops and deltas), type, angle, and location of minutiae points, ridge counts between pairs of minutiae, and location of pores. All these features contribute to fingerprint individuality. Although most individuality studies are based on minutiae representation, a few studies used Level 3 features as additional features and reported improved individuality. However, one should be careful when introducing Level 3 features as their permanence is still controversial. For example, based on direct photographs of fingers collected over a long period, Monson et al. (2019) claim that the size, shape, and presence of pores were not permanent.

Given a representation scheme and a similarity metric, there are two approaches for determining—the individuality of the fingerprints: *empirical* and *theoretical*. In the *empirical* approach, *representative* samples of fingerprints are collected and, using a *typical* fingerprint matcher (automatic or human), the accuracy of the matcher is calculated which provides an indication of the uniqueness of the fingerprint with respect to the matcher. Instead of collecting representative samples of the entire population, one could instead get an upper bound of matching accuracy by matching most genetically similar fingerprints, i.e., from identical twins (Jain et al., 2001, 2002; Lin et al., 1982). On the other hand, in a *theoretical* approach to individuality estimation, one needs to model all realistic phenomena affecting inter-class and intra-class fingerprint pattern variations. Given the similarity metric, one could then theoretically estimate the probability of a random correspondence. Theoretical approaches are often limited by the extent to which the assumed models conform to reality. In this chapter, we give a brief survey of the existing theoretical and empirical approaches to fingerprint individuality estimation. Section 8.4 is devoted to the problem of fingerprint persistence, which is another premise of fingerprint matching.

8.2 Theoretical Approach

8.2.1 Early Individuality Models

Early fingerprint individuality studies (as reviewed in Stoney and Thornton, 1986) are mainly theoretical approaches. These studies have typically focused on minutiae-based representations; some studies explicitly factored in fingerprint class (e.g., right loop, left loop, whorl, arch, tented arch, etc.) information. The minutiae type (e.g., ridge ending and ridge bifurcation), angle, and location of minutiae are the most commonly used features in these individuality studies. See Table 8.1 for a comparison of the features used in various fingerprint individuality models (including recent ones).

The use of the types of minutiae vary from one study to another: some studies used two minutiae types (ridge ending and bifurcation), whereas others (e.g., Osterburg, 1964;

Table 8.1 Fingerprint features used in different individuality models

Authors	Fingerprint features
Galton (1892)	Ridges, minutiae types
Pearson (1930, 1933)	Ridges, minutiae types
Henry (1900)	Minutiae locations, types, core-to-delta ridge count
Balthazard (1911) (cf. Stoney & Thornton, 1986)	Minutiae locations, two types, and two angles
Bose (1917) (cf. Stoney & Thornton, 1986)	Minutiae locations and three types
Wentworth and Wilder (1918)	Minutiae locations
Cummins and Midlo (1943)	Minutiae locations and types, core-to-delta ridge count
Gupta (1968)	Minutiae locations and types, fingerprint types, ridge count
Roxburgh (1933)	Minutiae locations, two minutiae types, two orientations, fingerprint and core types, number of positionings, area, fingerprint quality
Amy (1948) (cf. Stoney & Thornton, 1986)	Minutiae locations, number, types, and orientation
Trauring (1963)	Minutiae locations, two types, and two orientations
Kingston (1964)	Minutiae locations, number, and types
Osterburg et al. (1977)	Minutiae locations and types
Stoney and Thornton (1986)	Minutiae locations, distribution, orientation, and types, variation among prints from the same source, ridge counts, and number of alignments
Pankanti et al. (2002)	Minutiae locations, number, and angle
Zhu et al. (2007)	Minutiae locations, number, and angle
Chen and Moon (2008)	Minutiae locations, number
Su and Srihari (2010)	Minutiae locations, number, and angle
Lim and Dass (2011)	Minutiae locations, number, and angle

Osterburg et al., 1977) used as many as 13 types of events (empty cell, ridge ending, ridge bifurcation, island, dot, broken ridge, bridge, spur, enclosure, delta, double bifurcation, trifurcation, and multiple events). Some models included additional features (e.g., ridge counts Stoney, 1985; or sweat pores Roddy & Stosz, 1997) to determine the probability of occurrence of a particular fingerprint configuration. Most of the early individuality studies examined the distinctiveness of a portion/feature of the fingerprint. By assuming that events (e.g., placement of minutiae) are independent and identically distributed, these studies estimated the distinctiveness of the entire fingerprint (total pattern variation) by collating the distinctiveness in the features extracted from fingerprints (total feature variation). We refer to these total pattern variation-based fingerprint individuality estimates as

the probability of a fingerprint configuration. A summary of these studies is presented in Table 8.1.

The fingerprint individuality problem was first addressed by Galton in 1892 (Galton, 1892), who considered a square region spanning six-ridges in a given fingerprint. He assumed that, on average, a full 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 $1/2$, by looking at the surrounding ridges (see Fig. 8.1). Accordingly, the probability of a specific fingerprint configuration, given the surrounding ridges, is $(1/2)^{24}$. Galton 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

$$P(\text{Fingerprint Configuration}) = \frac{1}{16} \times \frac{1}{256} \times \left(\frac{1}{2}\right)^{24} = 1.45 \times 10^{-11}, \quad (8.1)$$

where $1/16$ is the probability of occurrence of a specific fingerprint type (such as arch, tented arch, left loop, right loop, double loop, whorl, etc.) and $1/256$ is the probability of occurrence of the correct number of ridges entering and exiting each of the 24 regions.

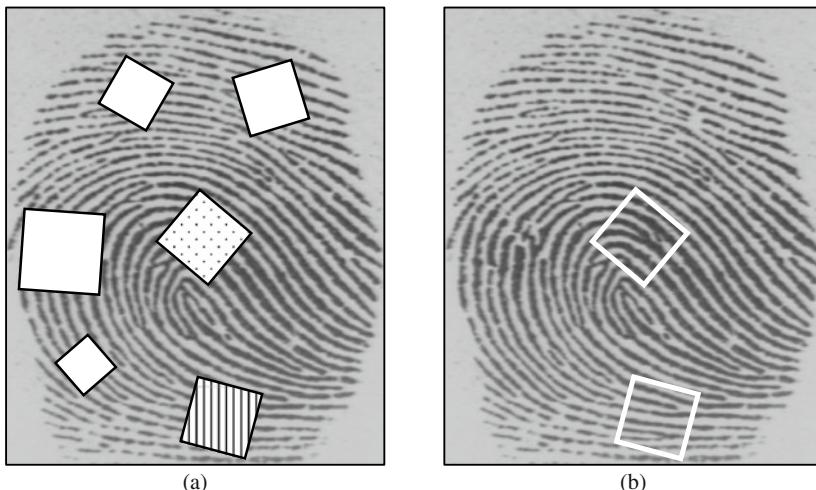


Fig. 8.1 Example of Galton's method of individuality estimation. Galton laid out enlargements of the fingerprints on the floor. He then dropped different sized squares such that they fell randomly on the enlarged fingerprint. Galton sought that size of paper square where he could correctly guess the ridge structure hidden underneath with a probability of $1/2$. For example, one can easily guess the ridges hidden by the six-ridge wide square marked with dots in **a** by looking at the surrounding ridges around the square. On the other hand, one cannot correctly guess the ridge structure hidden by the six-ridge wide square marked with vertical stripes in **a**. The hidden areas for these two squares are shown in **b**.

Equation (8.1) 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 (1933), Pearson (1930, 1933), and Kingston (1964) objected to Galton's assumption that the probability of occurrence of any particular ridge configuration in a six-ridge square is 1/2 and claimed that Eq. (8.1) grossly underestimates fingerprint individuality (i.e., overestimates the probability of occurrence). Pearson (1930, 1933) argued that there could be 36 (6×6) possible minutiae locations within one of Galton's six-ridge-square regions, leading to a probability of occurrence of a particular fingerprint configuration of

$$P(\text{Fingerprint Configuration}) = \frac{1}{16} \times \frac{1}{256} \times \left(\frac{1}{36}\right)^{24} = 1.09 \times 10^{-41}. \quad (8.2)$$

A number of subsequent models (Henry, 1900; Balthazard, 1911 (cf. Stoney & Thornton, 1986); Bose, 1917 (cf. Stoney & Thornton, 1986); Wentworth & Wilder, 1918; Cummins & Midlo, 1943; Gupta 1968) are interrelated and are based on a fixed probability p for the occurrence of a minutia. They compute the probability of a particular n -minutiae fingerprint configuration as

$$P(\text{Fingerprint Configuration}) = p^n. \quad (8.3)$$

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

- Henry (1900) chose $p = 1/4$ and added two to the number of minutiae n if the finger-print type and core-to-delta ridge count could be determined from the given (latent) fingerprint.
- Balthazard (1911) (cf. Stoney and Thornton, 1986) also set $p = 1/4$, under the assumption that there are four types of equally likely minutiae events: bifurcation to the right, bifurcation to the left, termination to the right, and termination to the left.
- Bose (1917) (cf. Stoney & Thornton, 1986) adopted $p = 1/4$, under the assumption that there are four possibilities in each square region of one ridge interval width in a fingerprint: a dot, a bifurcation, a ridge ending, and a continuous ridge.
- Wentworth and Wilder (1918) chose 1/50 as the value of p .
- Cummins and Midlo (1943) adopted the same value of p as Wentworth and Wilder (1918), but introduced a multiplicative constant of 1/31 to account for the variation in fingerprint pattern type.
- Gupta (1968) estimated the value of p as 1/10 for bifurcations and ridge endings, and 1/100 for the less commonly occurring minutiae types, based on 1,000 fingerprints. He also used a fingerprint type factor of 1/10 and correspondence in ridge count factor of 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 (1933) 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 n -minutiae fingerprint configuration to be

$$P(\text{Fingerprint Configuration}) = \left(\frac{C}{\mathcal{P}} \right) \times \left(\frac{Q}{R \times T} \right)^n, \quad (8.4)$$

where \mathcal{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, and $Q = 3.0$ 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 (1948) (cf. Stoney & Thornton, 1986) 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 and each reference point correspondence) increase the possibility of false association which is given by

$$P(\text{False Association}) = 1 - (1 - P(\text{Fingerprint Configuration}))^K. \quad (8.5)$$

Kingston's (1964) 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:

$$P(\text{Fingerprint Configuration}) = e^{-y} \times \frac{y^n}{n!} \times P_1 \times \left(\prod_{i=2}^n P_i \frac{(0.082)}{[S - (i-1)(0.082)]} \right), \quad (8.6)$$

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 in the i th minutia.

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 (1996) used an AFIS to extract minutiae from 977 fingerprint images 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 cluster configuration (five ridge endings and two bifurcations) as 2.25×10^{-5} .

Osterburg et al. (1977) divided fingerprints into discrete cells of size 1×1 mm. They computed the frequencies of 13 types of minutiae events (including an empty cell) from 39 fingerprints (8591 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 1.25×10^{-20} . Sclove (1979) 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 (1986) 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 (1986) 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), and number of comparisons performed with other fingerprints for identification.

Stoney's (1985) model is different from other models in that it attempts to characterize a significant component of pairwise minutiae dependence. Stoney (1985) and Stoney and Thornton (1986) 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 minutiae configuration and recursively estimated the probability of an n -minutiae configuration from the probability of an $(n - 1)$ -minutiae configuration and the occurrence of a new minutia of certain type/orientation at a particular distance/ ridge count from its nearest minutia within the $(n - 1)$ -minutiae configuration. The model also incorporated constraints due to connective ambiguity and due to minutiae-free areas. The model corrected for the probability of false association by accounting for the various possible linear orderings that 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

$$\begin{aligned} P(\text{False Association}) &= 1 - \left(1 - 0.6 \times (0.5 \times 10^{-3})^{(n-1)}\right)^{\lfloor n/5 \rfloor} \\ &\approx \frac{n}{5} \times 0.6 \times (0.5 \times 10^{-3})^{(n-1)} \end{aligned} \quad (8.7)$$

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 on the probability, and there are no constraints due to connective ambiguity, minutiae-free areas, or minutiae-free borders. Finally, it is (arbitrarily) assumed that one in every five minutiae can potentially serve as a starting point for a new search. 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 finger) was not sufficiently addressed. Their tolerances for minutiae position were derived from successive printings under ideal conditions and are far too low to be applicable in practical fingerprint comparisons.

The models discussed above (including Amy's model of false association due to multiple comparisons) focused 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 refer to the quantifications of fingerprint individuality that explicitly consider the intra-class variations as the probability of a random correspondence. Trauring (1963) was the first to concentrate explicitly on measuring the amount of detail needed to establish a correspondence between two prints from the same finger (intra-class variation) using an AFIS and observed that corresponding fingerprint features in impressions of the same finger 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 (ridge 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 from different fingers to be

$$P(\text{Random Correspondence}) = (0.1944)^n. \quad (8.8)$$

Stoney and Thornton's (1986) criticism of the Trauring model is that he did not consider ridge count, connective ambiguity, and correlation among minutiae location. Furthermore, 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 8.2.

Most of the early approaches to fingerprint individuality do not explicitly account for the sources of intra-class variability (see Sect. 4.1) in their models, and therefore, overestimate fingerprint individuality (give a smaller probability of random correspondence). This variability in multiple impressions of a finger manifests itself into (i) detection of spurious minutiae or missing genuine minutiae, (ii) displacement/disorientation (also called deformation) of the genuine minutiae, and (iii) transformation of the type of minutiae

Table 8.2 A comparison of the 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 ($G = 24$) as defined by Galton, 72 regions ($B = 72$) as defined by Osterburg et al., and has 36 minutiae on average ($n = 36$), we compute the probability of observing a given fingerprint configuration in the last column of the table. The probability of observing a fingerprint configuration with $n = 12$ and, equivalently, $G = 8$ and $B = 24$, is also 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 8.3)

Authors	P(Fingerprint Configuration)	Probability values for $n = 36, G = 24, B = 72$ ($n = 12$, $G = 8, B = 24$)
Galton (1892)	$(1/16) \times (1/256) \times (1/2)^G$	1.45×10^{-11} (9.54×10^{-7})
Pearson (1930, 1933)	$(1/16) \times (1/256) \times (1/36)^G$	1.09×10^{-41} (8.65×10^{-17})
Henry (1900)	$(1/4)^{n+2}$	1.32×10^{-23} (3.72×10^{-9})
Balthazard (1911) (cf. Stoney & Thornton, 1986)	$(1/4)^n$	2.12×10^{-22} (5.96×10^{-8})
Bose (1917) (cf. Stoney & Thornton, 1986)	$(1/4)^n$	2.12×10^{-22} (5.96×10^{-8})
Wentworth and Wilder (1918)	$(1/50)^n$	6.87×10^{-62} (4.10×10^{-22})
Cummins and Midlo (1943)	$(1/31) \times (1/50)^n$	2.22×10^{-63} (1.32×10^{-22})
Gupta (1968)	$(1/10) \times (1/10) \times (1/10)^n$	1.00×10^{-38} (1.00×10^{-14})
Roxburgh (1933)	$(1/1000) \times (1.5/24.12)^n$	3.75×10^{-47} (3.35×10^{-18})
Trauring (1963)	$(0.1944)^n$	2.47×10^{-26} (2.91×10^{-9})
Osterburg et al. (1977)	$(0.766)^B \cdot n! / (0.234)^n$	1.33×10^{-27} (1.10×10^{-9})
Stoney (1985)	$(n/5) \times 0.6 \times (0.5 \times 10^{-3})^{n-1}$	1.20×10^{-80} (3.50×10^{-26})

Table 8.3 Probabilities of a random correspondence obtained from the individuality model of Pankanti et al. (2002) for different sizes of fingerprint images containing 26, 36, or 46 minutiae. The entry (70, 12, 12, 12) corresponds to the 12-point guideline. The value of M for this entry was computed by estimating the typical fingerprint area manifesting 12 minutiae in a 500 dpi optical fingerprint scan

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

(connective ambiguity). This entails designing a similarity metric (matcher) that accommodates these intra-class variations. Furthermore, because most of the early models of individuality do not address the problems associated with the 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 seven matched minutiae between two fingerprints that have 18 and 37 minutiae, respectively?). This is a very important issue in an automatic fingerprint matching system where the feature extraction algorithms do not always provide the true minutiae and in matching latent prints to full prints. Although the likelihood of *detecting* false minutiae is significantly smaller in a manual fingerprint matching procedure than in automated systems, the manual procedure suffers from lack of consistency. The approach described in Pankanti et al. (2002), introduced in Sect. 8.2.2, not only explicitly models the situation of partial representational match, but also incorporate constraints on the configuration space imposed by intra-class variations (e.g., number of minutiae, minutiae position/orientation, image area) based on empirical estimates derived from the ground truth data marked by an expert on fingerprints obtained in a realistic environment.

8.2.2 Uniform Minutiae Placement Model

Pankanti et al. (2002) developed a simple fingerprint individuality model in an attempt to estimate the probability of a random correspondence between fingerprints. To make the model tractable, they made the following simplifying assumptions:

1. Only ridge endings and ridge bifurcation minutiae features are considered, because the occurrence of other minutiae types such as islands, dots, enclosures, bridges, double bifurcations, trifurcations, and so on is relatively rare. Because minutiae can reside only on ridges that follow certain overall patterns in a fingerprint, the minutiae angles are not completely independent of the minutiae locations. The statistical dependence between minutiae angles and locations is implicitly modeled.
2. A uniform distribution of minutiae in a fingerprint is assumed with the restriction that two minutiae cannot be very close to each other. Although minutiae locations are not uniformly distributed, this assumption approximates the slightly over-dispersed uniform distribution found by Stoney (1988).
3. Correspondence of a minutiae pair is an independent event and each correspondence is equally important. It is possible to assign a higher weight to spatially diverse correspondences compared to correspondences localized in a narrow spatial neighborhood.

4. Fingerprint image quality is not explicitly taken into account in the individuality determination. It is very difficult to reliably assign a quality index to a fingerprint because image quality is a subjective concept.
5. Ridge widths are assumed to be the same across the population and spatially uniform in the same finger. This assumption is justified because pressure variations could make non-uniform ridge variations uniform and vice versa.
6. The analysis of matching results of different impressions of the same finger binds the parameters of the probability of matching minutiae in two fingerprints from different fingers.
7. It is assumed that there exists one and only one (correct) alignment between the template and the input minutiae sets. It is assumed that a reasonable *alignment* has been established between the template and the input.

The model

Given an input fingerprint containing n minutiae, Pankanti et al. (2002) computed the probability that an arbitrary fingerprint (i.e., a template in the database) containing m minutiae will have exactly q corresponding minutiae with the input. The fingerprint minutiae are defined by their location, [x,y] coordinates, and by the angle of the ridge on which they reside, θ . The template and the input minutiae sets \mathbf{T} and \mathbf{I} , respectively, were defined as

$$\mathbf{T} = \{\{x_1, y_1, \theta_1\}, \{x_2, y_2, \theta_2\}, \dots, \{x_m, y_m, \theta_m\}\}, \quad (8.9)$$

$$\mathbf{I} = \{\{x'_1, y'_1, \theta'_1\}, \{x'_2, y'_2, \theta'_2\}, \dots, \{x'_n, y'_n, \theta'_n\}\}. \quad (8.10)$$

Under this simple model, a minutia i in the input fingerprint is considered as “corresponding” or “matching” to the minutia j in the template, if and only if

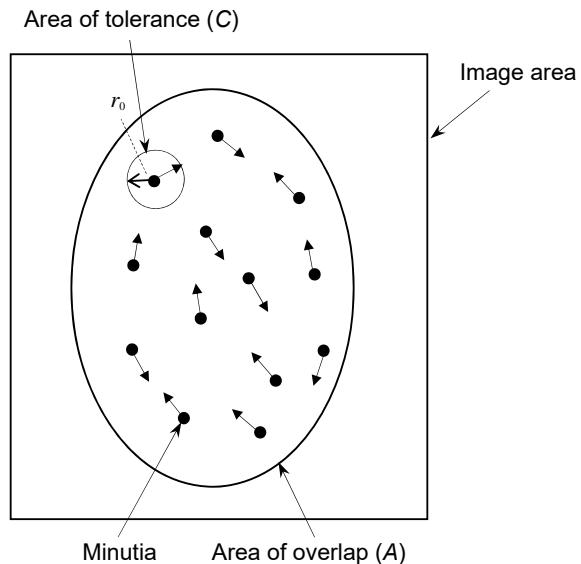
$$\sqrt{(x'_i - x_j)^2 + (y'_i - y_j)^2} \leq r_0, \text{ and} \quad (8.11)$$

$$\min(|\theta'_i - \theta_j|, 360^\circ - |\theta'_i - \theta_j|) \leq \theta_0, \quad (8.12)$$

where r_0 is the tolerance in distance and θ_0 is the tolerance in angle. Both manual and automatic fingerprint matches are based on some tolerance in minutiae location and angle to account for the variations in different impressions of the same finger (see Sect. 4.3.1). Equation (8.12) computes the minimum of $|\theta'_i - \theta_j|$ and $360^\circ - |\theta'_i - \theta_j|$ 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 (see Fig. 8.2). The probabilities that an arbitrary minutia in the input will match an arbitrary minutia in the template, only in terms of

Fig. 8.2 The area of the input fingerprint image that overlaps with the template and the input minutiae within the overlap area are shown. In addition, tolerance (in area) for minutia matching for one particular minutia is illustrated



location, and only in terms of angle, are given by Eqs. (8.13) and (8.14), respectively. Equation (8.13) assumes that $[x,y]$ and $[x',y']$ are independent and Eq. (8.14) assumes that θ and θ' are independent.

$$P\left(\sqrt{(x' - x)^2 + (y' - y)^2} \leq r_0\right) = \frac{\text{area of tolerance}}{\text{total area of overlap}} = \frac{\pi r_0^2}{A} = \frac{C}{A}, \quad (8.13)$$

$$P(\min(|\theta' - \theta|, 360 - |\theta' - \theta|) \leq \theta_0) = \frac{\text{angle of tolerance}}{\text{total angle}} = \frac{2\theta_0}{360}. \quad (8.14)$$

First, consider the scenario when only minutiae locations are matched; minutiae angles are introduced later in the formulation. If the template contains m minutiae, the probability that one minutia in the input will correspond to any of the m template minutiae is given by mC/A . Note that this and the subsequent location-based probability estimates are based on the assumption that the minutiae in fingerprints follow a slightly over-dispersed uniform distribution (Stoney, 1988); that is, only one template and one input minutia can occur in a single tolerance area (C). If this assumption is violated, the model becomes brittle and mC/A could actually become greater than one.

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 minutia has a correspondence (mC/A) and the second minutia does not have a correspondence $(A - mC)/(A - C)$. Thus, the probability that exactly one of the two input minutiae matches any of the m template minutiae is $2 \times (mC/A) \times (A - mC)/(A - C)$, as either the first input minutia alone may have a correspondence or the second input minutia 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

$$P(A, C, m, n) = \binom{n}{1} \left(\frac{mC}{A} \right) \left(\frac{A - mC}{A - C} \right). \quad (8.15)$$

The probability that there are exactly ρ corresponding minutiae, given n input minutiae, m template minutiae, the area of overlap (A), and area of tolerance (C) is

$$\begin{aligned} P(\rho|A, C, m, n) &= \underbrace{\binom{n}{\rho} \left(\frac{mC}{A} \right) \left(\frac{(m-1)C}{A-C} \right) \dots \left(\frac{(m-\rho-1)C}{A-(\rho-1)C} \right)}_{\rho \text{ terms}} \\ &\times \underbrace{\left(\frac{A-mC}{A-\rho C} \right) \left(\frac{A-(m-1)C}{A-(\rho+1)C} \right) \dots \left(\frac{A-(m-(n-\rho+1)C}{A-(n-1)C} \right)}_{n-\rho \text{ terms}}. \end{aligned} \quad (8.16)$$

The first ρ terms in Eq. (8.16) denote the probability of matching ρ 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. (8.16) by C , replacing A/C with M , and assuming that M is an integer (which is a realistic assumption because A is much greater than C), one can write the above equation in a compact form as (Rice, 1995)

$$P(\rho|M, m, n) = \frac{\binom{m}{\rho} \binom{M-m}{n-\rho}}{\binom{M}{n}}. \quad (8.17)$$

Equation (8.17) defines a hypergeometric distribution of ρ with parameters m , M , and n (Rice, 1995). 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 are 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 of M cards for the template fingerprint and an identical deck of M cards 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 ρ cards among the cards drawn is given by the hypergeometric distribution in Eq. (8.17) (Rice, 1995).

The above analysis considers a minutiae correspondence based solely on the minutiae location. Since minutiae patterns are generated by the underlying fingerprints which are smoothly flowing oriented textures, the orientations of neighboring minutiae points are strongly correlated. Further, the orientations of minutiae points are also correlated with their locations depending on the fingerprint type. 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.

To account for the dependence between two minutiae with orientations θ and θ' , let l be such that $P(\min(|\theta' - \theta|, 360^\circ - |\theta' - \theta|) \leq \theta_0) = l$ in Eq. (8.14). Given n input and m template minutiae, the probability of minutiae falling into *similar* positions can be estimated by Eq. (8.17). Once ρ minutiae positions are matched, the probability that q ($q \leq \rho$) minutiae among them have similar angles is given by

$$\binom{\rho}{q} (l)^q (1-l)^{\rho-q},$$

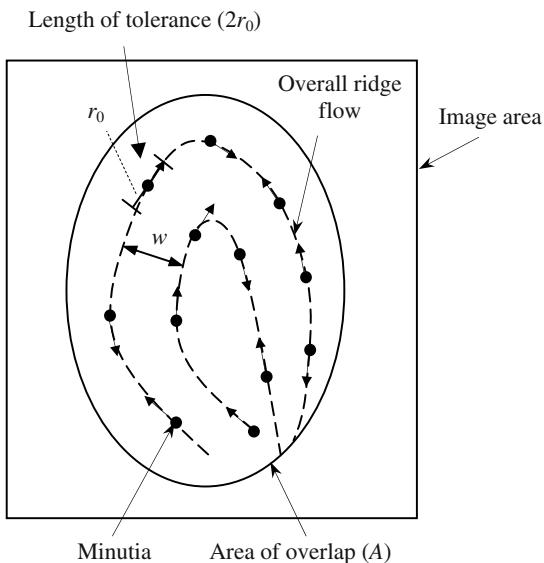
where l is the probability of two position-matched minutiae having a similar angle and $1-l$ is the probability of two position-matched minutiae having different angles. This analysis assumes that the ridge angle information/uncertainty can be completely captured by $P(\min(|\theta' - \theta|, 360^\circ - |\theta' - \theta|) \leq \theta_0)$. Therefore, the probability of matching q minutiae in both position as well as angle, given M , m , and n is

$$P(q|M, m, n) = \sum_{\rho=q}^{\min(m, n)} \left(\frac{\binom{m}{\rho} \binom{M-m}{n-\rho}}{\binom{M}{n}} \times \binom{\rho}{q} (l)^q (1-l)^{\rho-q} \right). \quad (8.18)$$

The above formulation has assumed that the minutiae locations are uniformly distributed within the *entire* fingerprint area. Since A is the area of overlap between the template and the input fingerprints, the ridges occupy approximately $A/2$ of the area with the other half occupied by the valleys. It is assumed that the number (or the area) of ridges across all fingerprint types is the same. Because the minutiae can lie only on ridges (i.e., along a curve of length A/w , where w is the ridge period), the value of M in Eq. (8.18) should, therefore, be changed from $M = A/C$ to $M = (A/w)/2r_0$, where $2r_0$ is the length tolerance in minutiae location (see Fig. 8.3).

The individuality model of Pankanti et al. (2002) has several parameters, namely: r_0 , l , w , A , m , n , and q . The value of l further depends on θ_0 . These parameters (r_0 , θ_0 , l , and w) further depend on the fingerprint scanner resolution. To compare the probabilities

Fig. 8.3 The area of the input fingerprint image that overlaps with the template and the input minutiae within the overlap area are shown. In addition, the overall ridge flow is shown and the tolerance (in length) for minutia matching for one particular minutia is illustrated



obtained from the theoretical model with the empirical results, the values of A , m , and n were estimated from two different databases as described in the next section. Interested readers can refer to the original paper for more details.

Experimental Evaluation

In order to evaluate their individuality model, Pankanti et al. (2002) used two fingerprint databases (called MSU_DB1 and MSU_VERIDICOM). The MSU_DB1 database contains fingerprint images of 167 subjects using an optical fingerprint scanner manufactured by Digital Biometrics, Inc. (image size = 508×480 , resolution = 500 dpi). Four impressions of the right index, right middle, left index, and left middle fingers for each subject are available that were captured over an interval of 6 weeks. The database contains a total of 2,672 ($167 \times 4 \times 4$) fingerprint images. The MSU_VERIDICOM database was collected following the same protocol, but using a solid-state capacitive fingerprint scanner manufactured by Veridicom, Inc. (image size = 300×300 , resolution = 500 dpi).

A large number of impostor matches (over 4,000,000) were generated using the automatic fingerprint matcher of Jain et al. (1997). The mean values of m and n for impostor matches were estimated as 46 for the MSU_DB1 database and as 26 for the MSU_VERIDICOM database from the distributions of m and n (Fig. 8.4a, b). The average values of A for the MSU_DB1 and the MSU_VERIDICOM databases are 67,415 pixels and 28,383 pixels, respectively. Pankanti et al. (2002) estimated the value of the overall effective area A 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 a common coordinate

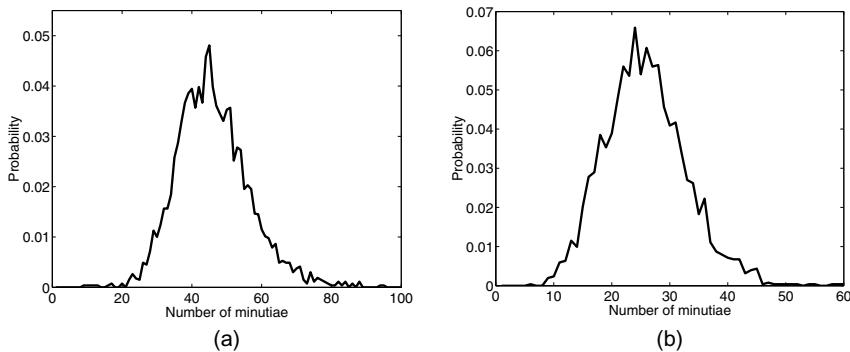


Fig. 8.4 Distributions of m and n for **a.** MSU_DB database, **b.** MSU_VERIDICOM database

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 matches performed in the database were pooled to obtain a distribution for A (see Fig. 8.5a, b). An arithmetic mean of the distribution was used to arrive at an estimate of A .

The probabilities of a random correspondence obtained for the different values of M , m , n , and q are given in Table 8.3. The values shown in Table 8.3 obtained based on the model of Pankanti et al. (2002) can be compared with values obtained from the other models in Table 8.2 for $m = 36$, $n = 36$, and $q = 36, 12$.

Typically, a match consisting of 12 minutiae points (the *12-point guideline*) is considered as sufficient evidence in many courts of law. Assuming that an expert can correctly match all the minutiae in a latent, a 12-point match with the full-print template (see

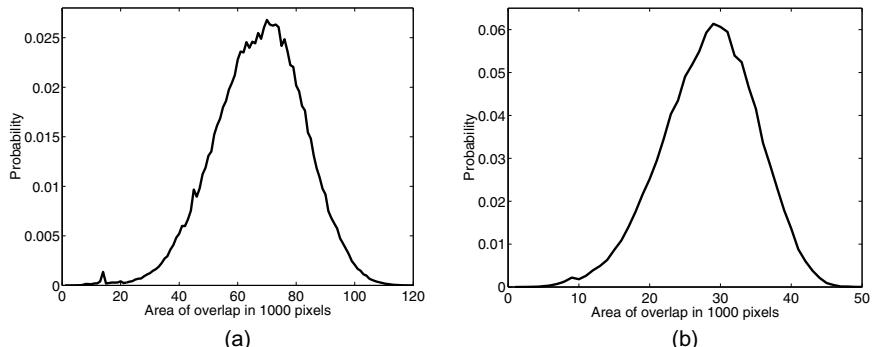


Fig. 8.5 Area of overlap between the two fingerprints that are matched based on the bounding boxes of the minutiae features for **a.** MSU_DB database, **b.** MSU_VERIDICOM database

Table 8.4 Effects of the fingerprint expert/matcher misjudgments in using the 12-point guideline. The source of error could be in underestimating the number of actual minutiae in the latent print (n) or overestimating the number of matched minutiae (q). The value of m is 12 for all entries in this table. The entry ($n = 12, q = 12$) represents the probability of a random correspondence when the 12-point guideline is correctly applied by a fingerprint examiner. Except for the ($n = 12, q = 12$) entry, all other entries represent incorrect judgments by the fingerprint expert to arrive at a decision that exactly 12 minutiae in the latent print matched 12 corresponding minutiae in the template print. For instance, the entry ($n = 14, q = 8$) in the table represents an estimate of the probability of a random correspondence due to two misjudgments by the examiner: the fingerprint examiner detected 12 minutiae in the latent print although there were in fact 14 minutiae in the latent print; that is, the examiner overlooked 2 latent print minutiae; furthermore, although he associated all 12 minutiae he detected in the latent print to the 12 minutiae in the template print, only 8 of those correspondences were indeed genuine correspondences (4 incorrect minutiae match judgments)

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

the first row, last column entry in Table 8.4) is an overwhelming amount of evidence, *provided* that there is no contradictory minutiae evidence in the overlapping area. The value of A was computed for 500 dpi fingerprint images from the minutiae density of 0.246 minutiae/mm² estimated by Kingston (1964) from 100 fingerprints; thus $M = 70$ was used for all the entries in Table 8.4. Because latent prints are typically of very poor quality, minutiae detection and matching errors are frequent. The effect of such errors on the probability of a random correspondence can be severe. For instance, two incorrect minutiae matches increase the probability of a random correspondence from 1.22×10^{-20} (entry $n = 12, q = 12$ in Table 8.4) to 1.96×10^{-14} (entry $n = 12, q = 10$ in Table 8.4) and ignoring two genuine minutiae present in the input (latent) print increases the probability from 1.22×10^{-20} (entry $n = 12, q = 12$ in Table 8.4) to 1.11×10^{-18} (entry $n = 14, q = 12$ in Table 8.4). Thus, a false minutiae match has significantly more impact than that of missing genuine minutiae in the input latent print.

Figure 8.6a, b show the distributions of the number of matching minutiae computed from the MSU_DB1 and MSU_VERIDICOM databases using the matcher of Jain et al. (1997), respectively. These figures also show the theoretical distributions obtained from the model of Pankanti et al. (2002) for the average values of M, m , and n computed from the databases. The empirical distribution is to the right of the theoretical distribution. This is because the theoretical model deviates from the Jain et al.'s (1997) matcher at several places. First, the theoretical model assumes that the “true” alignment between the

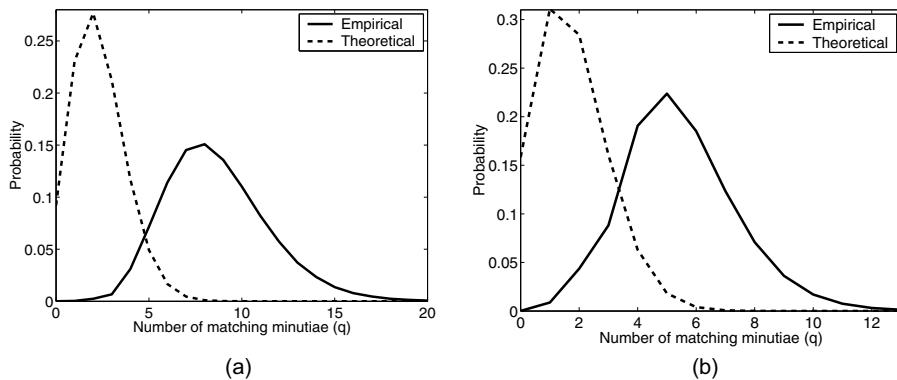


Fig. 8.6 Comparison of experimental and theoretical probabilities of random correspondence: **a.** MSU_DB database, **b.** MSU_VERIDICOM database

input and the template is known although the matcher estimates the alignment between the two fingerprints based on the minutiae information alone. For example, if there is only one minutia in the input fingerprint, the matcher will establish an alignment such that this minutia matches with a minutia in the template with a probability of 1. Thus, the theoretical probability estimate of (mC/A) for $n = 1$ is a gross underestimate for this matcher. In addition, the matcher seeks that alignment that maximizes the number of minutiae correspondences. Thus, it may find an alignment that is wrong but results in a large number of minutiae correspondences. Moreover, the matcher tests a large number of alignment hypotheses and, consequently, the probability of a random correspondence increases significantly according to Eq. (8.5). Second, the theoretical model assumes that two minutiae cannot be closer than the tolerance distance of $2r_0$ both in the input and the template fingerprints. However, the automatic matcher does not enforce this requirement and both the input and the template minutiae sets contain minutiae that are closer than the tolerance. This difference between the theoretical model and the automatic matcher becomes larger in the case of poor-quality fingerprint images where the matcher detects clusters of spurious minutiae. Finally, as explained in Table 8.4, any spurious minutia detected by the matcher increases the probability of a random correspondence.

Table 8.5 shows the empirical probabilities of matching 10 and 15 minutiae in the MSU_VERIDICOM and MSU_DB databases, respectively. The typical values of m and n were estimated from their distributions by computing the arithmetic means. The probabilities of a random correspondence for these values of m , n , and q are reported in the third column of Table 8.5 (note that Table 8.5 reports the probabilities of matching exactly q minutiae). The probabilities for matching “ q or more” minutiae are 3.0×10^{-2} and 3.2×10^{-2} for the MSU_VERIDICOM and MSU_DB databases, respectively; that is, they are of the same order of magnitude. The probabilities of the false match (FMR) obtained on these databases are consistent with those obtained on similar databases by several

Table 8.5 Probabilities of a random correspondence obtained from matching impostor fingerprints using an automatic matcher (Jain et al., 1997) for the MSU_VERIDICOM and MSU_DB1 databases. The probabilities given in the table are for matching “exactly q ” minutiae. The average values for A , m , and n are 28,383, 26, and 26 for the MSU_VERIDICOM database and 67,415, 46, and 46 for the MSU_DB1 database, respectively

Database	m, n, q	P(Random Correspondence)
MSU_VERIDICOM	26, 26, 10	1.7×10^{-2}
MSU_DB1	46, 46, 15	1.4×10^{-2}

other automatic fingerprint verification systems reported in the FVC2002 fingerprint verification competition (Maio et al., 2002). On the other hand, the performance claims by several fingerprint vendors vary over a large range (a false match rate of 10^{-9} to 10^{-3}) due to the different characteristics of the databases they use. The probabilities of a random correspondence from Pankanti et al. (2002) theoretical model obtained for different values of M , m , n , and q given in Table 8.3 are several orders of magnitude lower than the corresponding empirical probabilities given in Table 8.5.

8.2.3 Other Models

Pankanti et al. (2002) assumed a uniform distribution as the model on minutiae locations and angles to derive the probability of a random correspondence between a pair of fingerprints. The uniform model on fingerprint minutiae has several drawbacks. It is well known that fingerprint minutiae form clusters (see, for example, Stoney & Thornton, 1986). Further, minutiae locations in different regions of the fingerprint domain are observed to be associated with different region-specific minutiae angles. Also, minutiae that are spatially close tend to have similar angles. Empirical observations such as these need to be taken into account when eliciting reliable statistical models on fingerprint features. To alleviate the problem with the uniform distribution, several researchers proposed different statistical models of minutiae. Finite mixture models were developed to represent minutiae clusters in Zhu et al. (2007). Bayesian network was used to capture the distribution of minutiae along with dependence between them in Su and Srihari (2010). Inhomogeneous spatial point process was used to model characteristics of distribution of minutiae at different scales in Chen and Moon (2008), and Lim and Dass (2011).

8.3 Empirical Approach

Different from a theoretical approach, an empirical approach does not build a generative model for fingerprint features, nor does it derive a formula to estimate the probability distribution of matching scores (such as the number of matching minutiae) based on the statistical model of fingerprints. Instead, it first obtains a large number of matching scores computed by fingerprint matchers from real data, and then builds a statistical model based on the probability distribution of matching scores. Such a methodology is applicable for both fingerprint and other biometric traits.

Golfarelli et al. (1997) formulated the optimum Bayesian decision criterion for a biometric verification system; assuming the data distributions to be multi-normals, they derived two statistical expressions for theoretically calculating the false match and false non-match rates. By inferring the model parameters from real prototypes, they obtained a theoretical equal error rate of 1.31×10^{-5} for a hand-geometry-based verification system and of 2×10^{-3} for a face-based verification system.

Daugman (1999, 2015) proposed to model the distribution of Hamming Distance (HD) between different irises as a binomial distribution. The probability density function fitting the empirical scores is

$$f_0(x) = \frac{N!}{m!(N-m)!} p^m (1-p)^{(N-m)},$$

where x is the fractional HD score, and the binomial distribution describes the probability of Bernoulli trials occurring m times out of N trials with a probability p of any single trial. It is found that the FMR estimated by this fitting is close to the actual large-scale test of NIST. For example, at Hamming Distance threshold 0.29, the predicted FMR is 1 in 92 billion, while the measured FMR by NIST is 1 in 40 billion. He also considered the increase of FMR due to multiple alignments.

Compared to iris, fingerprint poses several challenges for modeling. First, rather than relying on absolute alignment like iris matching, fingerprint matching usually requires relative alignment. Second, the scoring methods for fingerprints are usually more complex and diverse. In fact, there is no widely accepted scoring method like Hamming distance for iris. For commercial fingerprint matchers, scoring is a trade secret. Statistical models based on one fingerprint matcher can neither allow a clear explanation between scores and features nor be directly generalized to other matchers. Despite these challenges, there are some researches on this topic.

Meagher et al. (1999) 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, namely, the intra-class variability. Furthermore, they assumed that the impostor and the

genuine distributions follow a Gaussian distribution and computed the probability of a random correspondence to be 10^{-97} . This model grossly underestimates the probability of a random correspondence because it does not consider realistic intra-class variations in impressions of a finger.

Egli et al. (2007) proposed to evaluate the matching scores of matcher for minutiae based on the assessment of likelihood ratios

$$LR = \frac{P(E|H, I)}{P(E|\bar{H}, I)},$$

where E is the observed evidence. H supports that the mark and the known fingerprint are homologous, while \bar{H} is the alternative, and I is any relevant background information that affects the possibility of evidence. The distribution of genuine matching scores and false matching scores can be thus obtained. The Weibull distribution and the Log-Normal distribution are, respectively, derived to model the genuine and false matching distributions. The matching scores and the variance of the matching scores of the genuine matching distributions increase with the number of minutiae, and the distributions in the tail are poorly fitted. In the comparison of repeated samples, it is found that the effect is only related to the size of samples. The genuine matching distribution is obtained from a large number of samples of the same finger, and the within-finger variability can only be evaluated from the minutiae of homologous fingerprints. Compared with the cross comparison of non-matching fingerprints, it is usually more difficult and complex to obtain and estimate a large number of samples of the same finger.

Nagar et al. (2012) proposed to measure the strength of fingerprint evidence using the non-match probability (NMP). NMP measures the probability that a non-mate decision made for the pair is correct, which is mathematically defined as

$$NMP(s) = P(I|s) = \frac{P(s|I)P(I)}{P(s|I)P(I) + P(s|G)P(G)},$$

where I and G correspond to non-paired/paired categories, respectively, and the relationship between the matching score s and its corresponding NMP value is called the NMP graph. The above equation applies the Bayes theorem for calculation.

When judging the matching degree of a pair of fingerprints manually, the determination is obviously affected by the amount of fingerprint information. For example, when considering two non-mate pairs of fingerprints both with the same low matching scores, we usually have more confidence that the pair with higher image quality is a mismatch. Therefore, this pair of fingerprints should probably be assigned a higher NMP value. Other factors that affect the amount of fingerprint information, such as the number of minutiae and the size of the fingerprints, should also have a similar effect on NMP calculation. Hence, they simulated a large number of fingerprints by cropping sub-images of different sizes, qualities, and number of minutiae. Due to the relatively large number of samples

at each point of the NMP curve, the obtained NMP curve is quite reliable. The extended definition of NMP is thus given by

$$\text{NMP}(\Phi_{f_1, f_2}) = P(I|\Phi_{f_1, f_2}),$$

where f_1 and f_2 correspond to the two fingerprints for comparison and Φ is the set of covariates that can be calculated from them. The matching score may not take into account the amount of image information, so other variables need to be controlled unchanged while covariates (size, number of minutiae, and image quality) are modeled into the NMP curve separately.

Nagar et al. (2012) proposed procedural steps to estimate the NMP values of fingerprint pairs, as shown in Fig. 8.7. First, a large number of latent fingerprints are simulated by cropping regions of different sizes, qualities, and number of minutiae. Secondly, the available set of print pairs are partitioned according to various fingerprint image characteristics (such as image quality and size), and the NMP curves for each partition are computed. An effective partitioning of the database can ensure that various critical covariates are considered and that each partition has a sufficient number of latent fingerprint pairs. The goodness of a partition can be computed on the basis of conclusiveness. Third, given a sample fingerprint pair, an appropriate NMP curve is selected according to the covariates used to partition the database. Fourth, the specific NMP value is selected from the corresponding NMP curve using the matching score of the given fingerprint pair. The size of the training database and the amount of computation required depend on the accuracy of the NMP values required in the trial. The whole process above is fully automatic without any human intervention.

Different density estimation methods are compared by calculating the bias and variance of NMP curves. Due to the large difference in the bias, they use the method based on

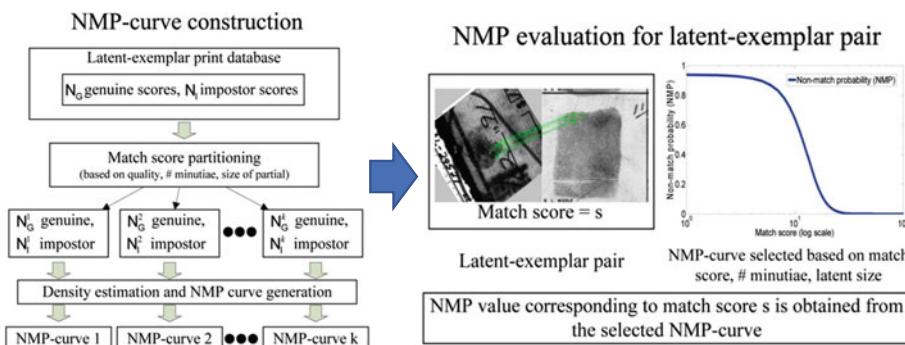


Fig. 8.7 Schematic diagram for computing the NMP value for a latent-exemplar fingerprint pair of interest based on NMP-curves obtained from a training database of latent-exemplar pairs (Nagar et al., 2012). © IEEE. Reprinted, with permission, from Nagar et al. (2012)

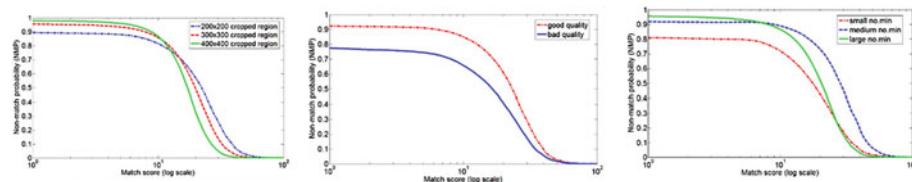


Fig. 8.8 NMP-curves for partial fingerprints of different sizes, quality levels, and numbers of minutiae (Nagar et al., 2012). © IEEE. Reprinted, with permission, from Nagar et al. (2012)

kernel density instead of parametric density estimation. A Gaussian kernel with different bandwidths is selected for the estimation, and the parameters used in density estimation are selected empirically so that the density estimates are as close to the corresponding match score histograms as possible.

In the trial of studying the influence of different fingerprint covariates on NMP curves, as shown in Fig. 8.8, the NMP curves with image size, image quality, and number of minutiae as a single variable are shown, respectively. We can conclude from the figure that fingerprints with a larger size have larger conclusiveness and stronger persuasion or discrimination. Similarly, the more minutiae, the more the NMP values deviate from 0.5, and the more decisive the matching scores are. Fingerprints with better quality are more convincing than bad ones. For low matching score pairs, NMP values with high-quality fingerprints are higher than low-quality ones, which is because if the quality is poor, it is more likely that genuine pairs would lead to low matching scores, thereby reducing the NMP values corresponding to low matching scores.

Neumann et al. (2006) calculated the Euclidean distance between fingerprints with their own matching method, which assessed the likelihood ratio by adopting a spatial modeling using radial triangulation and a probabilistic distortion model, and derived Tippett plots as performance indicators of the model. The advantage of this approach is that in addition to minutiae type and angle, the model incorporates the relative spatial relationships of the minutiae, which successfully simulates the within-finger variability for finger distortion. Moreover, the matching method is clear, while commercial fingerprint matchers used in other empirical studies are “black box”. The disadvantage is that the matching method is so simple that the matching performance is not as good as a state-of-the-art commercial matcher.

8.4 Persistence of Fingerprints

In addition to individuality, another fundamental premise for fingerprint matching is its persistence over time. In other words, does the fingerprint recognition accuracy degrade

over time? Compared to individuality, this topic has received less attention since it requires a large number of fingerprints for the same person collected over long time.

Yoon and Jain (2015) performed a longitudinal study of fingerprint recognition in order to examine the permanence of fingerprints according to fingerprint matchers. They derived 10-print cards of 15,597 individuals from the police database and collected them multiple times over a period of at least 5 years to obtain unbalanced and time unstructured longitudinal data. The data was used to investigate whether fingerprint recognition methods maintain high recognition accuracy as the time interval between fingerprints increases. A multi-level statistical model was used to analyze the similarity between fingerprints. The time interval of fingerprint comparison, age and gender of subjects, and fingerprint image quality were listed as covariates. They conducted linear modeling for intra-finger variability and inter-finger variability, respectively. Intra-finger variability took covariates and repeated measurements of subjects into account, while inter-finger variability mainly considered the population-mean tendency and deviations of subjects from the mean trend, and the errors were all assumed to be normally distributed. The matching scores were calculated by two commercial fingerprint matchers, where the genuine matching scores and impostor matching scores could both be obtained in the pairwise comparison. The binary judgment of genuine matching was thus made for whether the matching score of a fingerprint pair is larger than the threshold value. In the multi-layer model, the binary response was viewed as a Bernoulli trial with the probability of true acceptance, and the maximum likelihood estimation of the parameter was obtained by the iterative generalized least squares method under the premise that the residuals were normally distributed.

The following major conclusions are drawn:

- The hypothesis test for the slope of the linear model shows that with the increase of the comparison time interval between the two fingerprints, the genuine matching scores decrease significantly, while the change of the impostor matching scores is negligible. With the increase in the age of the subjects or the decrease in the quality of finger- print image, the genuine matching scores show a decreasing trend. Nevertheless, the probability of true acceptance is still close to 1, which is up to 12 years, the maximum time span of the dataset. However, if either of the two fingerprints is of poor quality, the uncertainty of the true acceptance probability becomes considerably large. The changes in impostor matching scores over time and the age of the subject are negligible. Thus, regardless of the time interval between two fingerprints, the probability of error acceptance is still close to 0.
- The influences of different covariates on the model of genuine matching scores are different. Time interval, subject age, and fingerprint image quality would lead to the change of genuine matching scores, while subject gender and race have little influence. With high-quality fingerprints from the same finger, the COTS matchers consistently provide high genuine matching scores. The results of single finger analysis are consistent with the results of 10-finger score fusion analysis.

8.5 Summary

There are two types of approaches for fingerprint individuality: the theoretical approach and the empirical approach. In the theoretical approach, a statistical model of fingerprints (usually minutiae) is built, and the probability of random matching is estimated. If such estimation is accurate, it is undoubtedly an ideal method for building the individuality model. However, there are large differences between existing theoretical approaches and real fingerprint matchers, including feature expression, fingerprint alignment, skin deformation processing, noise processing, calculation of matching scores, and so on. Some differences lead to underestimation of individuality, while others may lead to overestimation. With various factors mixed, it is generally impossible to judge the degree of deviation between the theoretical estimate and the real probability. In addition, it is quite difficult to quantitatively evaluate and compare different theoretical approaches. The empirical approach has no gap with the actual fingerprint matcher since it is based on the modeling of matching scores of a matcher on real data. But the biggest problem is that its performance depends on the performance of the specific matcher used. In latent fingerprint matching, fingerprint matchers are still far from perfect and are mainly used as sorting tools in practice.

One possible approach is to combine theoretical and empirical approaches to overcome each other's shortcomings. As the performance of fixed-length representation-based matchers approaches minutiae-based matchers (at least on certain databases), an individuality model based on fixed-length representation is also worth exploring.

References

- Amy, L. (1948). Recherches sur l'identification des traces papillaires. *Annales de Medecine Legale*, 28(2), 96–101.
- Balthazard, V. (1911). De l'identification par les empreintes digitales. *Comptes Rendus des Seances de l'Academie des Sciences*, 152, 1862–1864.
- Bose, H. C. (1917). *Hints on finger-prints with a telegraphic code for finger impressions*. Calcutta/Simla: Thacker Spink and Company.
- Champod, C., & Margot, P. A. (1996). Computer assisted analysis of minutiae occurrences on fingerprints. In J. Almog & E. Spinger (Eds.), *Proceedings of International Symposium on Fingerprint Detection and Identification*, (pp. 305). Israel National Police, Jerusalem
- Chen, J., & Moon, Y. S. (2008). The statistical modelling of fingerprint minutiae distribution with implications for fingerprint individuality studies. In *Proceedings of International Conferences on Computer Vision and Pattern Recognition (CVPR08)* (pp. 1–7).
- Cole, S. A. (2001a). What counts for identity? *Fingerprint Whorld*, 27(103), 7–35.
- Cole, S. A. (2001b). *Suspect identities: A history of fingerprint and criminal identification*. Cambridge, Harvard University Press.
- Cole, S. A. (2005). More than zero: Accounting for error in latent fingerprint identification. *The Journal of Criminal Law & Criminology*, 95(3), 985–1078.
- Cummins, H., & Midlo, C. (1943). *Fingerprints, palms and soles*. Dover.

- Daugman, J. (1999). Recognizing persons by their iris patterns. In A. K. Jain, R. Bolle, & S. Pankanti (Eds.), *Biometrics: Personal identification in a networked society*. Kluwer.
- Daugman, J. (2015). Information theory and the IrisCode. *IEEE Transactions on Information Forensics and Security*, 11(2), 400–409.
- Egli, N. M., Champod, C., & Margot, P. (2007). Evidence evaluation in fingerprint comparison and automated fingerprint identification systems - Modelling within finger variability. *Forensic Science International*, 167(2–3), 189–195.
- Galton, F. (1892). *Finger prints*. Macmillan.
- Golfarelli, M., Maio, D., & Maltoni, D. (1997). On the error-reject tradeoff in biometric verification systems. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 19(7), 786–796.
- Gupta, S. R. (1968). Statistical survey of ridge characteristics. *International Criminal Police Review*, 218(130).
- Haber, L., & Haber, R. N. (2004). Error rates for human latent fingerprint examiners. In N. Ratha, & R. Bolle (Eds.), *Automatic fingerprint recognition systems*. Springer.
- Henry, E. (1900). *Classification and uses of finger prints*. Routledge.
- Houck, M. M., & Siegel, J. A. (2009). *Fundamentals of forensic science* (2nd ed.). Academic Press.
- Jain, A. K., Hong, L., Pankanti, S., & Bolle, R. (1997). An identity authentication system using fingerprints. *Proceedings of the IEEE*, 85(9), 1365–1388.
- Jain, A. K., Prabhakar, S., & Pankanti, S. (2001). Twin test: On discriminability of fingerprints. In *Proceedings of International Conferences on Audio- and Video-Based Biometric Person Authentication* (3rd ed.).
- Jain, A. K., Prabhakar, S., & Pankanti, S. (2002). On the similarity of identical twin fingerprints. *Pattern Recognition*, 35(11), 2653–2663.
- Kingston, C. (1964). Probabilistic analysis of partial fingerprint patterns, Ph.D. Thesis, University of California.
- Lim, C. Y., & Dass, S. C. (2011). Assessing fingerprint individuality using EPIC: A case study in the analysis of spatially dependent marked processes. *Technometrics*, 53(2), 112–124.
- Lin, C. H., Liu, J. H., Ostenberg, J. W., & Nicol, J. D. (1982). Fingerprint comparison I: Similarity of fingerprints. *Journal of Forensic Sciences*, 27(2), 290–304.
- Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L., & Jain, A. K. (2002). FVC2002: Second fingerprint verification competition. In *Proceedings of International Conferences on Pattern Recognition* (16th ed.).
- Meagher, S. B., Buldowle, B., & Ziesig, D. (1999). 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.
- Monson, K. L., Roberts, M. A., Knorr, K. B., Ali, S., Meagher, S. B., Biggs, K., Blume, P., Brandelli, D., Marzioli, A., Reneau, R., & Tarasi, F. (2019). The permanence of friction ridge skin and persistence of friction ridge skin and impressions: A comprehensive review and new results. *Forensic Science International*, 297, 111–131.
- Nagar, A., Choi, H., & Jain, A. K. (2012). Evidential value of automated latent fingerprint comparison: An empirical approach. *IEEE Transactions on Information Forensics and Security*, 7(6), 1752–1765.
- Neumann, C., Champod, C., Puch-Solis, R., Egli, N., Anthonioz, A., Meuwly, D., & Bromage-Griffiths, A. (2006). Computation of likelihood ratios in fingerprint identification for configurations of three minutiae. *Journal of Forensic Sciences*, 51(6), 1255–1266.
- Newman, A. (2001). *Fingerprinting's reliability draws growing court challenges*. The New York Times.
- NRC. (2009). National Research Council. In *Strengthening forensic science in the United States: A path forward*. National Academies Press.

- Osterburg, J. W. (1964). An inquiry into the nature of proof: The identity of fingerprints. *Journal of Forensic Sciences*, 9(4), 413–427.
- Osterburg, J., Parthasarathy, T., Raghaven, T., & Sclove, S. (1977). Development of a mathematical formula for the calculation of fingerprint probabilities based on individual characteristic. *Journal American Statistic Association*, 72(360a), 772–778.
- Pankanti, S., Prabhakar, S., & Jain, A. K. (2002). On the individuality of fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8), 1010–1025.
- Pearson, K. (1930). *The life and letters of Francis Galton* (Vol. IIIA). Cambridge University Press.
- Pearson, K. (1933). Galton's work on evidential value of fingerprints. *Sankhya: Indian Journal of Statistics*, 1(50).
- Rice, J. A. (1995). *Mathematical statistics and data analysis* (2nd ed.). Duxbury Press.
- Roddy, A., & Stosz, J. (1997). Fingerprint features: Statistical-analysis and system performance estimates. *Proceedings of the IEEE*, 85(9), 1390–1421.
- Roxburgh, T. (1933). On evidential value of fingerprints. *Sankhya: Indian Journal of Statistics*, 1, 189–214.
- Saks, M. J., & Koehler, J. J. (2005). The coming paradigm shift in forensic identification science. *Science*, 309(5736), 892–895.
- Saltzman, J. (2005a). SJC bars a type of prints at trial. *The Boston Globe*.
- Saltzman, J. (2005b). Massachusetts Supreme Judicial Court to hear arguments on banning finger-print evidence. *The Boston Globe*.
- Sclove, S. L (1979). The occurrence of fingerprint characteristics as a two dimensional process. *Journal of American Statistical Association*, 74(367), 588–595.
- Stoney, D. A. (1985). A quantitative assessment of fingerprint individuality. Ph.D. Thesis, University of California.
- Stoney, D. A. (1988). Distribution of epidermal ridge minutiae. *American Journal of Physical Anthropology*, 77, 367–376.
- Stoney, D. A., & Thornton, J. I. (1986). A critical analysys of quantitative fingerprints individuality models. *Journal of Forensic Sciences*, 31(4), 1187–1216.
- Su, C., & Srihari, S. (2010). Evaluation of rarity of fingerprints in forensics. In *Proceedings of Advances in Neural Information Processing Systems* (pp. 1207–1215).
- Trauring, M. (1963). Automatic comparison of finger-ridge patterns. *Nature*, 197, 938–940.
- Wentworth, B., & Wilder, H. H. (1918). *Personal identification*. R.G. Badger.
- Yoon, S., & Jain, A. K. (2015). Longitudinal study of fingerprint recognition. *Proceedings of the National Academy of Sciences*, 112(28), 8555–8560.
- Zhu, Y., Dass, S. C., & Jain A. K. (2007). Statistical models for assessing the individuality of fingerprints. *IEEE Transactions on Information Forensics and Security*, 2(3), 391–401.



Securing Fingerprint Systems

9

Abstract

This chapter discusses the security issues and countermeasures that can be used to build secure fingerprint recognition systems. After an introduction to the different types of attacks that can be potentially launched, the chapter presents methods for (fraudulently) obtaining fingerprint data and their countermeasures. The focus is on presentation attack instruments (e.g., spoof fingerprints) and related presentation attack detection (PAD) techniques. Template protection techniques are then reviewed by focusing on feature transformations, biometric cryptosystems, and feature adaptions. Recent approaches such as homomorphic encryption of fixed-length representations are introduced, and the chapter concludes with a discussion on current challenges and open issues.

Keywords

Security • Attacks • Threat models • Presentation attack • Presentation attack detection • Spoof fingerprint • Altered fingerprint • Template protection • Biometric cryptosystems • Homomorphic encryption

Invited Chapter by Karthik Nandakumar, Mohamed Bin Zayed University of Artificial Intelligence.

9.1 Introduction

The primary purpose of employing a fingerprint recognition system is to provide a non-repudiable mechanism for establishing or verifying the identity of an individual. However, just like any other real-world system, fingerprint systems are susceptible to failures. But what exactly constitutes a fingerprint system failure? While the answer to this question depends on the application in which the fingerprint system is deployed, four common classes of failures associated with fingerprint systems are: (i) intrusion, (ii) denial-of-service, (iii) repudiation, and (iv) function creep. An illustration of these fingerprint system failures is presented in Fig. 9.1.

In a vast majority of applications, fingerprint systems are employed to control access to a physical or logical asset (e.g., a building or a mobile phone) or a service (e.g., a bank account). In such access control applications, the fingerprint system is analogous to a facility that is secured with a lock. The most common security failure in these applications is an *intrusion*, where an unauthorized entity gains illegitimate access to the protected facility. After gaining access, the adversary may either modify some data (e.g., issue a

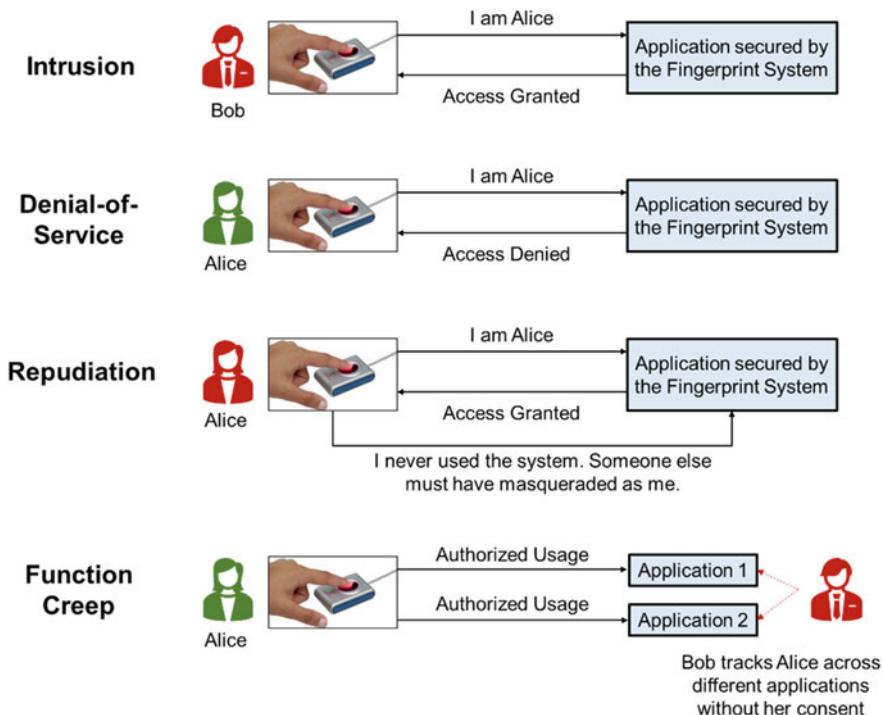


Fig. 9.1 Four major classes of failures associated with fingerprint systems are intrusion, denial-of-service, repudiation, and function creep

withdrawal of all the money in a bank account), access privileged data (e.g., customers' private information), or exploit the access to carry out further illegitimate activities.

There are many possible ways in which an intrusion threat can be launched. These are commonly referred to as *attack vectors*, which are discussed in detail in Sect. 9.2. The most direct mechanism for an adversary to gain entry into the facility is to break open the lock. In the case of fingerprint systems, this is akin to presenting a non-mated fingerprint and hoping the system will produce a false match error. In negative recognition applications (e.g., screening at international border control), this could involve presenting (possibly altered) fingerprints to trigger a false non-match error. Usually, the sturdier the lock, the higher is the perceived level of security. Similarly, a fingerprint system is perceived to be more secure if it has a very low false match rate. Note that a false match error is due to the intrinsic failure of a fingerprint system to recognize fingerprints correctly and does not require much effort from the adversary. Hence, this type of attack vector is also known as *zero effort attack*.

However, even if a facility is equipped with the strongest possible lock, it is still possible to break into the facility. For example, instead of trying to break the lock, a burglar may steal the genuine key from the owner, clandestinely clone a genuine key and use the duplicate key to open the lock, or coerce the owner at gunpoint to gain entry. Alternatively, the adversary may leave the lock untouched, but still force his way through by breaking open the door, making a big hole in the wall, or smashing the windows. Thus, a stronger lock does not necessarily mean better security and other attack vectors also need to be seriously considered.

One issue peculiar to biometrics and fingerprints is related to "revocation". In traditional password/cryptographic systems, if a certain password or key is compromised, it can be put on a "revocation list" to prevent its future use and a new one can be issued to the user. If a fingerprint is compromised, although it can be put on a revocation list, a new one cannot be issued (a different finger may still be used but unlike passwords and keys, we have only a limited number of fingers). As a result, if an adversary gets access to the fingerprint data of an individual, he may be able to intrude again and again. Further, since the individual may have enrolled the same finger in many different systems, an adversary who successfully intrudes that user's account in one system (say the weakest one) may then be able to intrude his/her account in a different system.

Apart from intrusion, other types of fingerprint system failures are also possible. In the context of access control applications, ensuring that legitimate or authorized users can access the resources protected by the fingerprint system is an equally important objective alongside preventing intrusion by adversaries. Thus, if an attacker manages to damage the lock (fingerprint system) to such an extent that legitimate users are no longer able to gain access to the facility, it also constitutes a system failure. This type of failure is generally referred to as *denial-of-service* (DoS). In some mission critical applications (e.g., international border control), such a lock out may be devastating. In other applications, it can cause a very poor user experience and eventually drive the users towards abandoning

the system altogether. The DoS could be complete (everyone, every time) or partial. It could be caused by high intrinsic error rate of fingerprint systems (e.g., failure to acquire or false rejects) or caused by an active adversary (e.g., by sabotaging the power supply or overloading the system with bogus matching or communication requests, or damaging a fingerprint sensor, etc.). One of the reasons why adversaries may launch DoS attack is to force the use of a fallback mechanism (say a password system), which may then be easier for the adversary to intrude (rather than the fingerprint system).

While intrusion and denial-of-service are serious security failures that have the potential to undermine users' trust, they are not unique to fingerprint or biometric systems. Other authentication systems, as well as general cryptosystems, are also vulnerable to such failures. For example, Anderson (1994) studied the technical aspects of fraud involved in using ATMs and found that most of the failures were due to poor design and lack of control by system administration. What differentiates a fingerprint system (or in general, any biometric system) from other security systems is the promise of non-repudiation. The fundamental premise of fingerprint recognition is that fingerprints are unique and persistent and cannot be lost, shared, or stolen. Due to this strong link between the person and his/her fingerprint, fingerprint systems guarantee that an individual being granted access is indeed who he/she claims to be. When this guarantee is violated, it gives rise to a *repudiation* failure, which occurs when a legitimate user denies using the system after having accessed it. It must be emphasized that repudiation is an indirect security failure: corrupt users can deny their actions only by highlighting the fact that fingerprint systems are vulnerable to intrusion attacks.

Finally, the uniqueness of fingerprints also poses a new type of failure known as *function creep*. In this case, the adversary exploits the fingerprint system designed for a specific purpose to serve another completely unintended purpose. For instance, fingerprint data provided to a bank for enabling account access may be used in conjunction with finger-print information from an immigration database to track the travel history of a person. While function creep is more of a threat to the privacy of enrolled individuals, it is a consequence of failing to secure the biometric/fingerprint data of individuals. Hence, it is also an indirect security failure and making fingerprint systems robust to intrusion attacks can mitigate the likelihood of function creep.

It must be emphasized that irrespective of the installed security system, no system is absolutely secure or foolproof. Given the right opportunity and plenty of time and resources, any security system can be broken. But public confidence and acceptance of fingerprint systems will depend on the ability of system designers to guard against all possible security failures. Hence, any fingerprint system deployment must be preceded by a careful definition of a *threat model*, which outlines what needs to be protected and from whom. Threat models are almost always tied to the expected attacks (e.g., resources available, intent, and expertise of the attacker). Unless a threat model is clearly defined for a system, it is very difficult to decide if the proposed security solution is adequate. Depending upon the threat model of an application, an adversary may invest varying

degrees of time and resources in launching an attack. For example, a fingerprint system deployed in a critical application such as border control can be expected to have a higher risk of being attacked compared to a fingerprint system used for unlocking a mobile phone. Similarly, in a remote and unattended application that requires recognition from a remote server, a hacker may have the opportunity and plenty of time to mount numerous attacks or even physically violate the integrity of a client system.

In this chapter, we will focus our discussion only on the security issues and solutions that are specific and unique to fingerprint recognition systems. Ideally, a holistic view of system security is essential but due to the vast differences in application contexts and threat models, we focus primarily on intrusion threats. For a successful intrusion, an adversary needs to first obtain fingerprint data of an authorized user either with the cooperation of the user or covertly and then inject it into the authentication system through presentation attacks. We will discuss these mechanisms in detail along with techniques that have been proposed to counter the above threats. It is expected that one or more of these technological components will be useful in preparing a holistic security solution for a practical application with a given threat model.

9.2 Threat Model for Fingerprint Systems

The first step in analyzing the security of fingerprint systems is defining the threat model, which identifies the various threat agents and attack vectors. A taxonomy of attacks that can be mounted on fingerprint systems is shown in Fig. 9.2. Prior to venturing into a discussion on threat agents and attack vectors, it must be emphasized that a fingerprint system may fail due to its own *intrinsic limitations*. As discussed earlier, false match errors made by a fingerprint system enable zero-effort intrusion attacks. Depending on the application, false non-match errors and failure to enroll/acquire fingerprints of genuine users could lead to either denial-of-service (in access control applications) or intrusion (in negative recognition applications such as airport screening). However, these intrinsic limitations can be addressed by developing more robust, accurate, and easy-to-use fingerprint systems based on new sensors (Chap. 2), invariant representation methods (Chap. 3), and effective matching algorithms (Chap. 4). Hence, we do not discuss solutions to mitigate such intrinsic limitations in this Chapter.

Fingerprint systems may also fail to deliberate manipulation by threat agents or adversaries, who could be either insiders and/or external entities. *Insider attacks* often exploit the vulnerabilities introduced by human interactions with the fingerprint system. This category of attacks includes scenarios where an authorized user himself turns malicious and intentionally attempts to circumvent the system or is directly or indirectly used by an external adversary to subvert the system. Note that insiders could be authorized users of a fingerprint system or system administrators (super-users). On the other hand, *external*

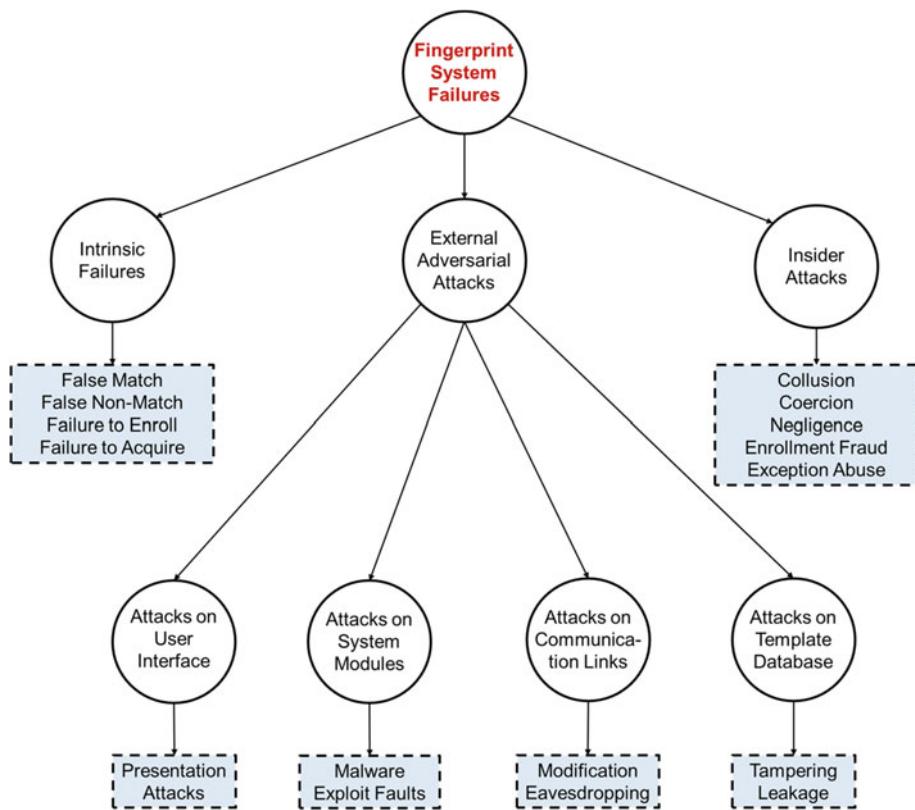


Fig. 9.2 A taxonomy of attacks that can be mounted on fingerprint systems

adversarial attacks typically rely on exploiting the loopholes in the infrastructure components of a fingerprint system including the user interface (sensor), system modules (feature extractor and matcher), interconnections between the system modules, and the template database.

9.2.1 Insider Attacks

Human interactions with the fingerprint system can be exploited in the following five ways to circumvent the intended operations of the system:

Collusion: Collusion refers to the scenario where an authorized user turns intentionally malicious and attacks the fingerprint system either individually or in cooperation with an external adversary (possibly in return for monetary gain). For example, if the external adversary has a friend who is an authorized user, he may simply request the authorized

user to provide her fingerprint data. The authorized user may either help the adversary in obtaining a valid fingerprint image or simply grant the adversary access to her account (or prop the door open in a physical access control application). This is, in fact, the most popular method featured in the media to showcase the ease of breaking a fingerprint system. In reality, such a scenario poses little threat in most applications. Possible safeguards against such an attack are mechanisms to enforce responsible behavior among authorized users through proper training, rigorous monitoring, and enforcing penalties for non-conformance with the rules. A more serious threat is when system administrators, who have control over most modules of a fingerprint system, attempt to modify the functioning of a system. It is extremely difficult to guard against such an attack.

Coercion: The only difference between collusion and coercion is the intention of the legitimate user. While collusion refers to the case where the legitimate user acts based on his own free will, coercion occurs when the legitimate user is forced by the adversary (e.g., through blackmail or threats of physical violence) to disabuse the fingerprint system. Again, there is no specific technological countermeasure to prevent this attack.

Negligence: This attack is also similar to collusion except that there is no explicit cooperation between the adversary and the authorized user. Typical examples include failure of authorized users to properly log out of an online system or leaving a door open after physical access. Such negligent behavior by authorized users can be exploited by the adversary to gain illegitimate access to the protected resources. Apart from periodic training and constant reminders to authorized users about the best practices to be followed, little can be done to prevent such attacks.

Enrollment Fraud: The Achilles heel of most fingerprint systems is their reliance on existing identity management systems for enrollment. In many applications, enrollment of users requires them to produce legacy identity credentials such as passport, driver's license, identity card, or birth certificate along with their fingerprints. An adversary can exploit this loophole and enroll himself into the fingerprint system illegitimately by producing falsified credentials. Thus, the integrity of the fingerprint system is constrained by the integrity of the enrollment process. Two types of failure can happen if the enrollment process is not foolproof: the same individual may obtain multiple identities and multiple individuals can be enrolled under the same (shared) identity. While the first type of failure can be a serious issue in applications like welfare disbursement, the second type of failure is problematic in applications where non-repudiation is a key objective (e.g., border control).

A possible solution to stop an individual from obtaining multiple identities is to match the fingerprints of a user requesting enrollment against the fingerprints of all existing enrolled users to detect a duplicate identity even before the requesting user is enrolled into the system. This process is called *de-duplication* and it has been demonstrated to work well in large-scale applications such as India's Aadhaar project with approximately 1.3 billion identities. The limitations of this approach are high computational cost, inability

to perform real-time enrollment, and the need to collect fingerprints from multiple fingers of an individual to achieve the required low false positive and false negative identification rates.

The second problem of multiple individuals sharing the same identity can be mitigated through careful supervision of the enrollment process by a human operator. This is the reason why most applications require the user to physically present herself at some dedicated facility to complete the enrollment process. In the absence of such supervision, the enrollment process can be compromised in a couple of ways. It has been recently demonstrated that most fingerprint recognition systems are vulnerable to attacks based on *double-identity* fingerprints, which are fake fingerprints generated by combining the features from two different fingers (Ferrara et al., 2017). These double-identity fingerprints have a high chance (about 90% chance of success at a false match rate of 0.1%) of being falsely matched to fingerprints from both the fingers. Moreover, such fake fingerprint patterns are realistic enough to fool human examiners. Another mode of attack becomes feasible in fingerprint recognition systems that allow the enrollment of multiple fingers. In this scenario, the different fingers may come from different individuals. This often happens in application scenarios such as smartphone unlock, where multiple members of a family or a group of friends may enroll their fingers with the same identity, thereby allowing them shared access to a protected resource, thereby violating the non-repudiation principle.

Exception Abuse: One of the thorny issues in most biometric systems is the lack of universality of the biometric trait. Invariably, in any fingerprint system, there will be some individuals who cannot provide their fingerprint due to physical disability or issues related to their skin condition (e.g., manual laborers with worn-out fingerprints). Similarly, there may be scenarios where the user may be temporarily unable to provide his fingerprints (e.g., wearing gloves, fingers with bandage) for authentication. To avoid denial-of-service in such exceptional cases, most fingerprint systems are designed to have a fallback mechanism that relies on other credentials like secrets and tokens. The drawback is that a motivated adversary can deliberately trigger this exception handling procedure and attempt to exploit the vulnerabilities of the fallback mechanism. One way to mitigate this problem is to deploy multimodal biometric systems that rely on different biometric traits (e.g., fingerprint, face, iris) and allow the user to choose the most appropriate biometric trait depending on the circumstances.

9.2.2 External Adversarial Attacks

External adversarial attacks primarily rely on attacking various infrastructure components of a fingerprint system, which includes hardware and software components such as sensor, feature extractor, matcher, template database, and decision module along with the communication links between these components. External adversarial attacks are arguably more

dangerous than insider attacks due to their high scalability (affecting a large proportion of users) and susceptibility to remote adversaries. Hence, there is an extensive body of literature that discusses the various vulnerabilities in fingerprint systems to such external attacks. Ratha et al. (2001) considered the typical architecture of a biometric system and identified eight specific *attack vectors* or *points of attack* (see Fig. 9.3). Cukic and Bartlow (2005) extended this framework by holistically looking at the application encapsulating the fingerprint system and identified 20 potential attack points with 22 possible vulnerabilities. Roberts (2007) presented a revised model to analyze the risk of attacks on biometric systems by focusing on three main dimensions of attacks: (i) threat agents, (ii) threat vectors, and (iii) system vulnerabilities. Jain et al. (2008) used a compact *fishbone* model to summarize the biometric system vulnerabilities. Fish-bone model captures the relationship between causes and effects of failures. They listed four causes, namely, intrinsic, non-secure infrastructure, administrative abuse, and biometric overtess and two effects, namely, denial of service and intrusion. More recently, the ISO/IEC 19989–1/2/3 (2020) standard on the criteria and methodology for security evaluation of biometric systems has been published and Part 1 of this standard summarizes the vulnerabilities of fingerprint systems within a common framework.

Most of the attack vectors discussed in the literature can be grouped under four broad categories: (i) attacks on the user interface, (ii) attacks on the software modules, (iii) attacks on the communication links, and (iv) attacks on the template database. It must be emphasized that these attack vectors per se are not unique to fingerprint systems, because they also afflict traditional knowledge and token-based identity management systems. In fact, the attacks on software modules and communication links of a fingerprint system are no different from similar attacks on password-based authentication systems. For instance, an adversary can potentially replace any software component of a fingerprint system with malware or Trojan horse program, which disguises itself as the true software component but generates false results (such as a fingerprint image, fingerprint features, match

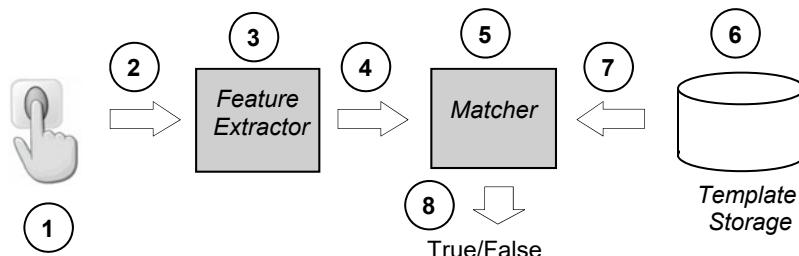


Fig. 9.3 High-level architecture of a fingerprint recognition system that illustrates the eight possible attack points identified by Ratha et al. (2001). These eight attack points can, in turn, be grouped into four broad categories: attacks at the user interface (1), attacks on the software modules (3 and 5), attacks on the communication links (2, 4, 7, and 8), and attacks on the template database (6)

score, or decision) desired by the attacker. The adversary may also exploit coding bugs or software faults to potentially circumvent the system. Similarly, the adversary can snoop, modify, or replay messages exchanged between the various components by the fingerprint system by eavesdropping or gaining control of the communication channels. Attacks on software modules can be mitigated by following secure coding and software development practices (McGraw, 2006). Communication channels can be secured using standard cryptographic techniques such as encryption, digital signatures, timestamp, challenge-response, etc. (Schneier, 1996). Moreover, both these attack vectors can also be countered by designing closed fingerprint systems, which are discussed in Sect. 9.7.

On the other hand, attacks on the user interface of a fingerprint system and fingerprint template database have certain distinctive characteristics, which are not observed in other knowledge-based authentication systems. For example, the primary challenge for the attacker in a knowledge-based authentication system is to guess the authorized user's secret correctly. Once the secret is known, presenting the secret at the user interface and gaining access to the system is often a trivial task. However, in a fingerprint system, both the mechanism for obtaining the fingerprint data of an individual and presenting the acquired fingerprint data at the user interface pose peculiar challenges. While a detailed discussion of the ways to acquire fingerprint data is presented in Sects. 9.3, 9.4 describes the ways in which the fingerprint data can be presented back to the system and Sect. 9.5 discusses the corresponding countermeasures. Furthermore, cryptographic tools such as hashing and encryption are typically used to protect password databases. However, as we will see in Sect. 9.6, these techniques cannot be directly applied to protect fingerprint template databases. This has led to the emergence of a new class of fingerprint template protection techniques.

9.3 Methods of Obtaining Fingerprint Data and Countermeasures

Let us first revisit some of the limitations of knowledge-based authentication so that the readers can better appreciate the corresponding challenges in fingerprint systems. The fundamental premise of knowledge-based authentication is that the authentication secret is known only to the authorized user. However, most users set their passwords based on words or numbers that they can easily remember, such as names and birthdays of family members, favorite movie or music star, and dictionary words. This makes the passwords easy to crack by guessing or a simple brute force dictionary attack. Although it is advisable to keep different passwords for different applications, most people use the same password across multiple applications. If a single password is compromised, it may open many doors. Long and random passwords are more secure but harder to remember, which prompts some users to write them down in accessible locations (e.g., as notes on a smartphone or computer). Strong (difficult to remember) passwords also result in more system help desk calls for forgotten passwords. Cryptographic techniques such as encryption can

provide very long passwords that are not required to be remembered but that are in turn protected by simple passwords, thus defeating their purpose. Furthermore, an attacker needs to break the password of only one employee to gain access to a company's Intranet and thus a single weak password compromises the overall security of the system. Therefore, the security of the entire system is only as good as the weakest password (weakest link). Finally, when a password is shared with a colleague, there is no way for the system to know who the actual user is.

The promise of fingerprint recognition is that fingerprints are significantly more difficult to copy, share, and distribute than passwords and tokens. Fingerprints cannot be lost or stolen and fingerprint-based recognition requires the person to be present at the point of authentication. It is difficult to forge fingerprints and unlikely for a user to repudiate having used the system. In short, no other security technology besides biometrics/fingerprints provides such a strong link between a physical person and an action taken by the person. While all these claimed advantages are theoretically true, one must also question these claims from a practical viewpoint. First and foremost, it turns out that copying or stealing someone's fingerprint data is indeed possible in practice. It is already known that biometric data in general is *personal*, but not *private/secret*. For example, in this age of internet and social media, face images and videos of individuals are widely published and accessible. While other biometric traits such as fingerprints and iris are somewhat more difficult to acquire covertly, there exist many ways for an adversary to obtain fingerprint data of a specific target user. For example, the adversary can:

- Lift a latent fingerprint left on physical surfaces that the user has touched.
- Extract the fingerprint from a high-resolution photo posted by the user on the Internet.
- Guess the fingerprint data by *hill climbing*.
- Gain direct access to the fingerprint data of the user from the template database.

It must be emphasized that all the above approaches pertain to obtaining the fingerprint data of a chosen target user. It may also be possible for the adversary to make an intrusion attempt by just injecting a generic fingerprint image sampled from a public-domain fingerprint database or generated synthetically. Such an attempt will be classified under a zero-effort attack, which can be easily thwarted by employing a fingerprint system with a very low false match rate. This greatly reduces the incentive for an adversary to select this mode of attack. Suppose a fingerprint matcher is operating at a false match rate (FMR) of 0.001%, then it will take, on average, 100,000 intrusion attempts with generic fingerprint data to succeed. However, there are applications such as unlocking a smartphone, where the false match rate cannot be set arbitrarily low in order to avoid significant inconvenience to genuine users. Furthermore, fingerprint sensors used in smartphones typically have small surface area and capture only partial fingerprints. Consequently, users are allowed to enroll many partial impressions as well as multiple fingers. This creates an

opportunity for adversaries to create what are known as “MasterPrints”, which can successfully match with a large number of fingerprint samples pertaining to multiple users. Roy et al. (2017) demonstrated that such generic MasterPrints can be used to impersonate a large number of users in fingerprint systems using both optical and capacitive fingerprint sensors.

Now, we describe the methods to obtain the fingerprint data of specific target individuals.

9.3.1 Lifting Latent Fingerprints

To lift a latent fingerprint of a specific user, the adversary needs to have knowledge about the whereabouts of that user. The adversary follows the targeted subject, obtains access to surfaces/objects that she has touched, lifts a latent fingerprint from it, and scans it into a fingerprint image. Possible techniques for capturing a latent fingerprint impression have already been discussed in Sect. 2.2. Unlike fingerprint images captured using live-scan fingerprint sensors, fingerprint impressions created by touching most surfaces are typical of poor quality because they are incomplete, wrapped around irregular surfaces, or partially smudged by the finger slipping. To lift a latent fingerprint from difficult surfaces, much expertise and advanced sensing technology may be needed (see Sect. 2.2). Even with the right tools and expertise, most of the lifted fingerprints are expected to be too poor in quality to be matched successfully by an automated system. There is also no economy of scale that the adversary can exploit since a single latent fingerprint cannot be used to launch attacks against multiple users. Additionally, the threat of an attacker obtaining such fingerprint data is fairly low in remote applications. So, obtaining latent fingerprints is complicated, expensive, and of limited use. While there is no specific countermeasure to stop an adversary from lifting a latent fingerprint, the threat is low in most applications. Nevertheless, this approach still remains a concern in some applications where the potential rewards for a successful intrusion are very high.

9.3.2 Extracting Fingerprints from High-Resolution Photos

This is a more recent threat fueled by the rapid advent of Internet, social media, and high-resolution digital cameras (including smartphones). High-resolution photos of many individuals get posted every day on the Internet. If these photos contain an unobstructed view of the fingerprints of an individual (e.g., while flashing a V-sign), it becomes feasible to extract the fingerprint patterns of the person from the photo (also see Sect. 2.3.1). For example, media reports indicate that a politician’s fingerprints were surreptitiously obtained by a German hacker in 2014 from high-resolution photographs of the politician captured at a distance using off-the-shelf digital cameras (Hern, 2014). Significant

advancements have also been made in the processing of “fingerphotos” captured using smartphone cameras and it has been clearly established that such fingerphotos contain sufficient information to be successfully matched against fingerprints captured using traditional contact-based fingerprint sensors (Priesnitz et al., 2021). While the fingerphotos in the above study were captured with the cooperation of the user (see Fig. 9.4), further improvements in both camera technology and processing algorithms will soon make it possible to extract good quality fingerprints even from finger photos captured under non-cooperative and covert settings. One of the countermeasures proposed to mitigate this threat is the concept of a wearable jamming pattern (Echizen & Ogane, 2018), which obfuscates fingerphotos, but does not affect the operation of a traditional fingerprint sensor.

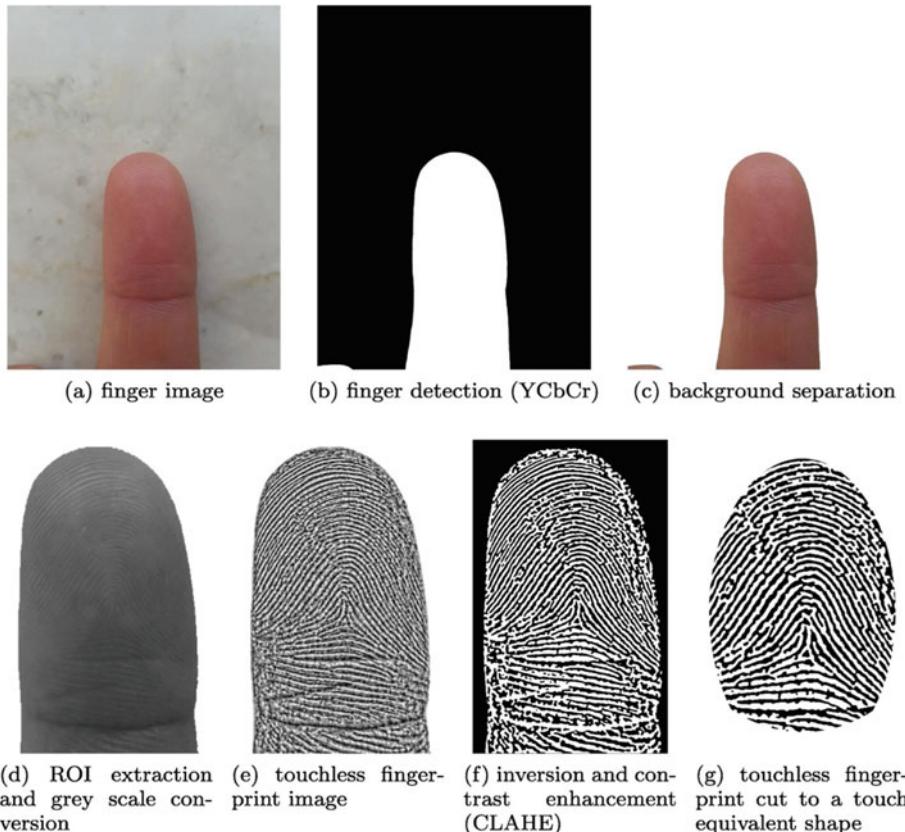


Fig. 9.4 Typical workflow involved in extracting the fingerprint pattern from a high-resolution photograph. Image reproduced from Priesnitz et al. (2021)

9.3.3 Guessing Fingerprint Data by Hill Climbing

If an adversary can inject a synthetically generated fingerprint image (or feature set) into the fingerprint system and if the matcher returns a score, then the adversary can iteratively optimize the process of generating synthetic fingerprint images to increase his chance of intrusion. Soutar (2002) was the first to describe such a hill climbing strategy in the context of a fingerprint recognition system. Hill climbing can be used to optimize the synthetic generation of fingerprint image or feature set. In this case, while the generated fingerprint data is not identical to the enrollment template, it is sufficiently similar to fool the matcher into yielding a decision of match. The hill climbing software iteratively modifies subsequent synthetically generated fingerprint data such that the match score is higher than in the previous iteration (i.e., the fingerprint image or feature set being synthesized moves towards becoming more similar to the template).

A countermeasure to hill climbing attack was also proposed by Soutar (2002). He proposed increasing the granularity of the fingerprint match score returned by the fingerprint matcher. He argued that if the match scores are granular (e.g., reported in steps of 10 for a match score in the range [0,100]), then the hill climbing method would require a sufficiently large number of attempts before observing a change in score and thus the total number of attempts required for the match score to exceed the system threshold would become prohibitively large. In the limiting case, the system may not output match scores at all and provide only the match/non-match decisions. In this case, the number of required attempts is determined by the false match rate, provided synthetic fingerprints that follow the real fingerprint distribution can be generated.

9.3.4 Stealing Fingerprint Data from the Template Database

One of the straightforward methods of obtaining the fingerprint data of an individual is to hack into template database of a fingerprint system that the person is enrolled in. This could be done either by colluding or coercing the system administrator or by completely bypassing the security rings around the database. In fact, the fingerprint template does not need to come from the same application being attacked. Rather it can be obtained from any fingerprint system that the target user is enrolled in. The fingerprint data obtained by the adversary may be in the form of a fingerprint feature set/template or fingerprint image. It has long been argued that if a fingerprint system uses fingerprint feature sets or templates in proprietary formats, the system is protected from the injection of fingerprint data as the adversary does not know its format. Similarly, claims have been made since the feature sets or templates are a highly compressed mathematical representation of the original fingerprint images, it is not possible to reconstruct back the fingerprint image given the template. However, it turns out that both these arguments lack sufficient merit.

If an attacker has access to a fingerprint image, he can easily convert it to fingerprint feature set or template by using a feature extractor (which does not have to be the same one used by the fingerprint system he intends to intrude; it is sufficient that it uses the same representation). Secrecy of a proprietary template coding (or interface) could be easily broken by an attacker with adequate resources. In fact, it is becoming customary for fingerprint recognition systems to adopt standard minutiae formats such as ISO/IEC 19794–2 (2011), and ANSI/NIST-ITL 1–2011 (2015) as well as standard APIs (Soutar, 2004). Therefore, it is reasonable to expect that a fingerprint image can be converted to a feature set or template irrespective of the coding scheme or format used by the target system.

When it comes to converting a fingerprint feature set or template back to fingerprint image, it is true that an exact reversal is not possible because some information is surely lost during feature extraction. However, a “close enough” replication that will fool an automated fingerprint recognition system is still possible. Several researchers including Hill (2001), Ross et al. (2007), Cappelli et al. (2007), Feng and Jain (2011), Li and Kot (2012), and Cao and Jain (2015) have shown that it is possible to reconstruct good quality fingerprint images using only the minutiae templates. Figure 9.5 shows examples of fingerprint images reconstructed only from fingerprint minutiae data based on some of the above methods. Such “close enough” fingerprint images can be used for injecting into the fingerprint system with very high success rate. For example, the reconstruction approach of Cao and Jain (2015) obtained an average successful attack rate of more than 95% even when the reconstructed fingerprint is matched against a different impression of the same finger, provided both the original fingerprint (used to extract the template) and the new impression are of good quality.

In case the fingerprint recognition system uses non-minutiae-based representation, the enrollment template (and feature sets) may actually contain the whole fingerprint image, perhaps in compressed form, as specified by the standard finger image data format ISO/IEC 19794–4 (2011). Further, standard finger pattern spectral data formats such ISO/IEC 19794–3 (2006) are specified to contain block-wise wavelet coefficients derived from the fingerprint image and again it is straightforward to reverse a fingerprint image from them.

9.3.5 Countermeasures for Protecting Fingerprint Data

Among the possible ways available for gleaning the fingerprint data of individuals, directly obtaining the data from the template database poses the most serious threat because it be done covertly and is a highly scalable approach for the adversary. The most straightforward method to protect the templates in an enrollment database is to save them in encrypted form using standard (and proven) cryptographic techniques (e.g., Advance Encryption Standard [AES] algorithm). Different encryption keys can be used

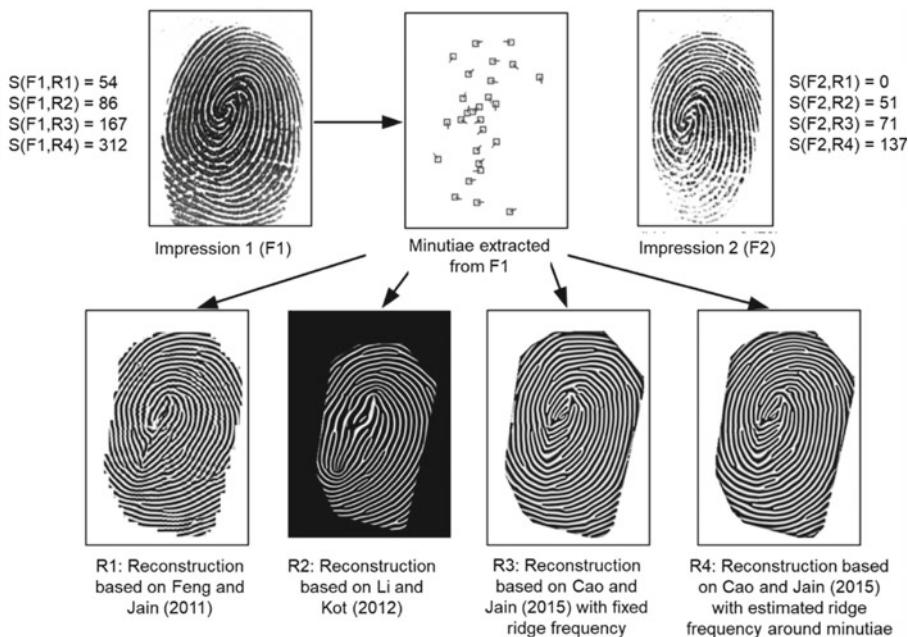


Fig. 9.5 Four examples of fingerprint image reconstruction from minutiae template. It can be observed that the reconstructed images (bottom row) match well not only with the original impression (F1) from which the minutiae template is obtained but also with a different impression (F2) of the same finger. Here, $S(x,y)$ represents the similarity between fingerprint impressions x and y estimated by a commercial fingerprint recognition software, where match scores greater than 50 indicate a very high-confidence match. © IEEE. Reprinted, with permission, from Cao and Jain (2015)

in different applications such that templates from one application cannot be interchanged with the templates of another application. If an encryption key is ever compromised, templates can be re-encrypted with a new cryptographic key. However, there is a critical issue with this approach; unlike password comparison, conventional fingerprint matching cannot be performed in the encrypted domain, and the templates need to be decrypted before the matching stage. This is because standard cryptographic algorithms transform even a tiny difference in the input space into huge differences in the output space, and since multiple impressions of the same finger can be quite different from each other, it is impossible to match them in the encrypted domain. The necessity to decrypt the template introduces a security weakness, since an adversary can access the memory where the template is decrypted and/or try to steal the decryption key. Once a key is compromised and templates are “stolen”, the templates cannot be revoked and reissued.

One way to protect the enrollment templates is to store them on tamper-resistant secure hardware (Sect. 9.7) and do not permit them to leave the secure hardware boundary. There

is a lot of interest in these methods and in their practical implementations such as match-on-card and system-on-a-chip systems (Grother et al., 2007). In fact, the Touch ID and Face ID systems on Apple iPhones follow this approach and store the templates within a secure enclave on a dedicated chip (see chapter on Touch ID and Face ID security in Apple 2021). One of the advantages of these techniques is that the template storage is distributed across the system users. For example, a user is always in the possession of his enrollment template residing in the secure enclave. Avoiding central storage also prevents an adversary to exploit economies of scale. However, storing templates on secure hardware has some drawbacks (e.g., accuracy drop, extra cost, etc.) and is very difficult in applications requiring central management or operating in identification mode. For example, some applications allow users to access their accounts from multiple access points without re-enrollment and this requires centralized template management. Therefore, there is a need for technologies to protect fingerprint templates even if the decryption key is compromised. Neither should the adversary be able to link identities across different applications if he gains access to template storage. A class of techniques, called *template protection techniques*, have been developed to provide such countermeasures. We discuss these techniques in Sect. 9.6.

Since in many cases, it is not possible to prevent the attacker from obtaining fingerprint data of individuals, the burden of security often shifts to techniques that prevent the injection of the acquired data into the system. One of the general countermeasures to prevent injection of fingerprint data against a specific user account is to “lock” the system after a small number of non-match decisions, say three, have occurred within a short period of time. This could help in the case where the adversary is using generic fingerprint data as it is expected that the adversary would need a large number of intrusion attempts before achieving success. The adversary may resort to injecting the same generic fingerprint data against a large number of accounts instead of injecting repeatedly against a single user but this countermeasure of limiting the number of attempts still provides some level of protection. However, if the adversary has access to the fingerprint data of a specific target individual, restricting the number of access attempts is unlikely to be beneficial in thwarting an attack. In this scenario, the only solution is to prevent the adversary from injecting illegitimate data into the system. Note that this injection can happen either at the user interface level or through the communication channels between the various modules of a fingerprint system. We have already seen that the communication links can be secured to a great extent using standard cryptographic tools. Hence, we will now focus on securing fingerprint systems against attacks that inject data at the user interface (or sensor) level, which are generally referred to as fingerprint presentation attacks.

9.4 Presentation Attacks

The ISO/IEC 30107–1 (2016) standard defines presentation attacks as the *presentation to the biometric data capture subsystem with the goal of interfering with the operation of the biometric system*. These attacks can be realized through a number of methods (see Fig. 9.6), which are called as *presentation attacks instruments* (PAI) in the above ISO standard. The most natural countermeasure to prevent presentation attacks is to add a hardware or software module to the sensor that detects the presence of an attack before the fingerprint captured by the sensor is accepted for further processing. Such a module is known as *fingerprint presentation attack detection* (PAD).

Presentation attack instruments in fingerprint systems include the use of (i) *gummy fingers* (Matsumoto et al., 2002), i.e., fabricated finger-like objects with accurate imitation of another individual's fingerprint ridge-valley structures, (ii) *2D or 3D printed fingerprint targets* (Cao & Jain, 2016; Arora et al., 2016; Engelsma et al., 2018), (iii) *altered fingerprints* (Yoon et al., 2012), i.e., intentionally tampered or damaged real fingerprint patterns to avoid identification, and (iv) *cadaver fingers* (Marasco & Ross, 2015). Among the various presentation attack instruments, fingerprint spoofs (i.e., gummy fingers and printed targets) are the most common and easiest instruments to launch presentation attacks.

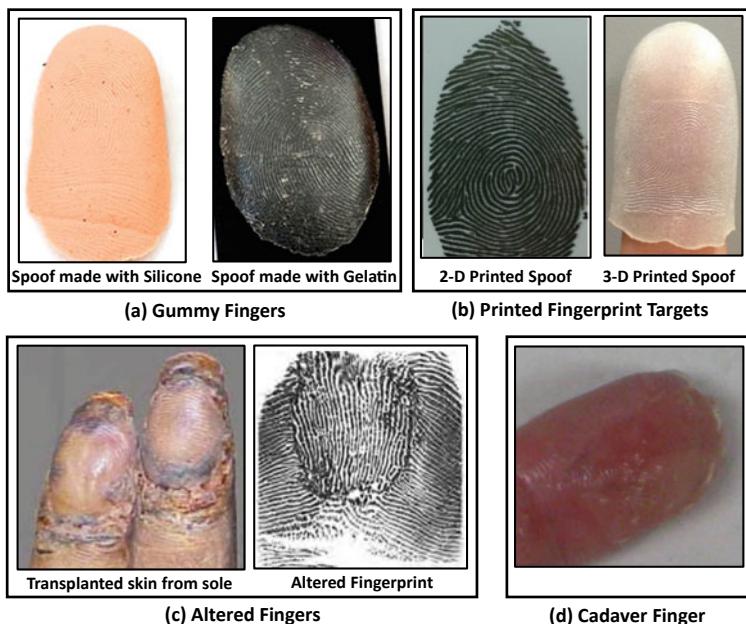


Fig. 9.6 Fingerprint presentation attacks can be realized using **a** gummy fingers, **b** 2D or 3D printed fingerprint targets, **c** altered fingers, or **d** cadaver fingers. © IEEE. Reprinted, with permission, from Chugh et al. (2017)

9.4.1 Fingerprint Spoofs

Fingerprint spoofs can be created with a multitude of fabrication processes ranging from basic *molding* and *casting* to utilizing sophisticated 2D and 3D printing techniques. It has been reported that commonly available and inexpensive materials, such as gelatin, silicone, Play-Doh, etc., can be utilized to fabricate high fidelity fingerprint spoofs, which are capable of bypassing a fingerprint recognition system (see Fig. 9.7). For instance, in March 2013, a Brazilian doctor was arrested for using spoof fingers made of silicone to fool the biometric attendance system at a hospital in Sao Paulo (BBC News, 2013). In another incident, in Sept. 2013, shortly after Apple released iPhone 5 s with in-built TouchID fingerprint technology, Germany's Chaos Computer Club (CCC, 2013) hacked its capacitive sensor by utilizing a high-resolution photograph of the enrolled user's fingerprint to fabricate a spoof fingerprint with wood glue. In July 2016, researchers at Michigan State University unlocked a fingerprint secure smartphone using a 2D printed fingerprint spoof to help police with a homicide case (Korkzan, 2016). In March 2018, a gang in Rajasthan, India, was arrested for spoofing the biometric attendance system, using glue casted in wax molds, to provide proxies for a police entrance exam (Vidyut,

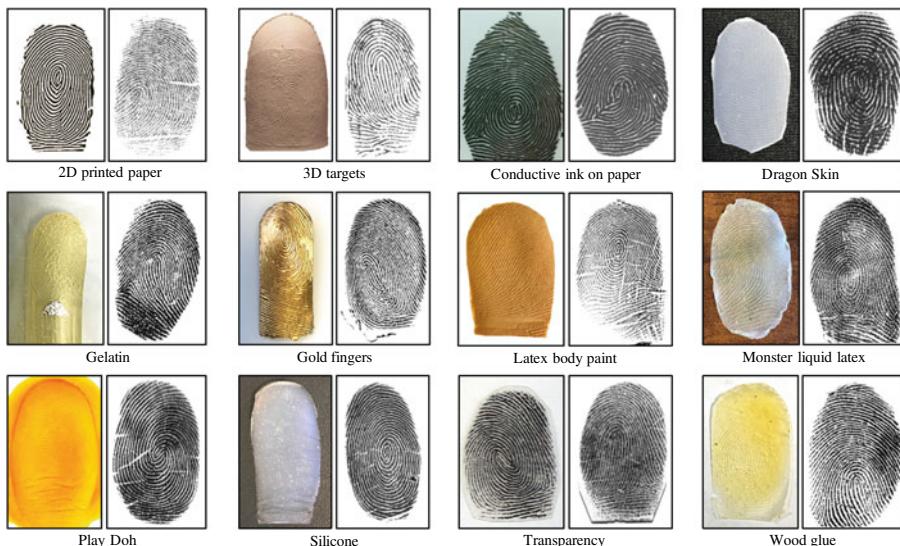


Fig. 9.7 Several commonly available materials can be utilized to create fingerprint spoof attacks. In this figure, fake fingers generated from these materials are shown along with grayscale fingerprint images captured using an optical sensor. The physical artifacts and the grayscale fingerprint images do not necessarily correspond to the same finger. © IEEE. Reprinted, with permission, from Chugh and Jain (2021)

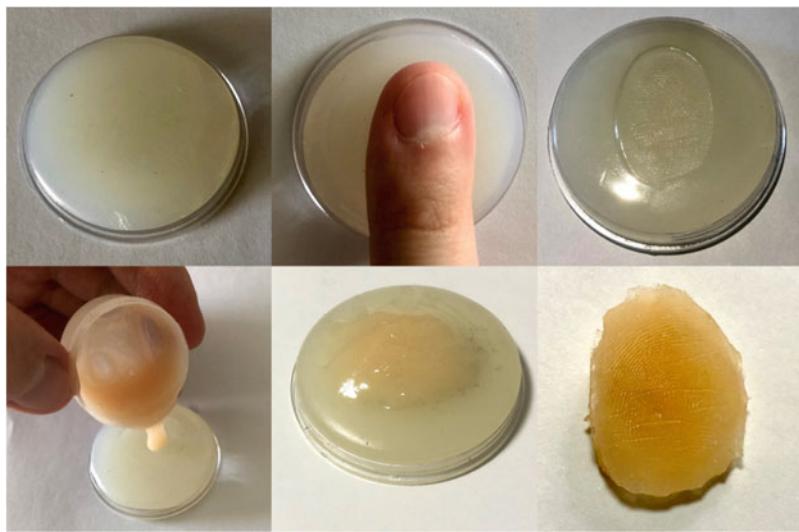


Fig. 9.8 Illustration of a cooperative process to create a fake gummy finger using molding and casting. Paraffin wax is used to create the mold and silicone is used as the casting material

2018). It is likely that a large number of these attacks are never detected and hence not reported.

Fabrication of a spoof can be a *cooperative* process (see Fig. 9.8), where the real fingerprint owner is involved and a live finger serves as one of the inputs to the process, or a *non-cooperative* process (see Fig. 9.9), where a fingerprint image (acquired using one of the methods described in Sect. 9.3) serves as the starting point to create a fake replica. Figure 9.10 shows an example of cooperative fabrication where the ridges are clearly visible on both the mold and spoof.

9.4.2 Altered Fingerprints

Another form of presentation attack is intentional fingerprint alteration, which leads to what is known as *altered fingerprints* (see Figs. 9.6 and 9.11). This type of presentation attack may happen in border control and law enforcement applications, where the adversary attempts to hide his true identity. Unlike gummy fingers, altered or obfuscated fingerprints are real fingers whose ridge structure has been severely altered by abrading, burning, cutting, or performing surgery on fingertips. As shown in Fig. 9.11, different types of alteration procedures would result in different fingerprint degradation. Based on the changes made to friction ridge patterns, altered fingerprints are categorized into three types: *obliteration*, *distortion*, and *imitation* (Yoon et al., 2012). *Obliteration* includes

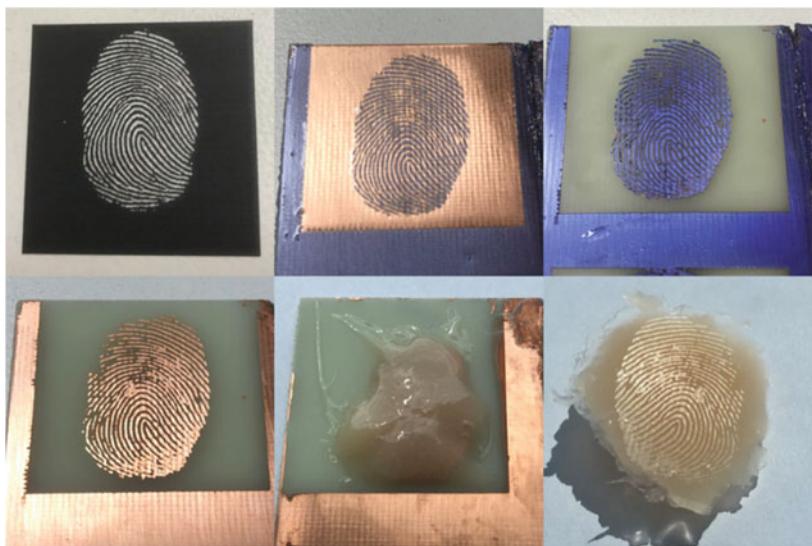


Fig. 9.9 Illustration of the non-cooperative process to create a fake replica using a residual fingerprint on printed circuit board (PCB)



Fig. 9.10 An example of cooperative fabrication where the mold (left side) is made of dental material and the spoof (central image) of PlayDoh. On the right, a fingerprint image produced by the spoof. Image courtesy of Stephanie Schuckers and David A. Yambay

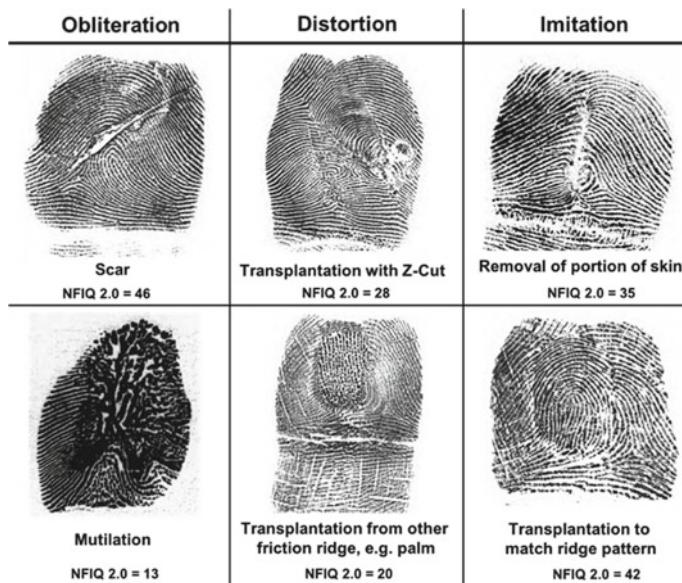


Fig. 9.11 Three different forms of altered fingerprints, namely obliteration, distortion, and imitation. Their quality scores (based on NFIQ 2.0) are also presented, where a score close to 100 implies high quality, and a score of 0 implies low quality. © IEEE. Reprinted, with permission, from Tabassi et al. (2018)

abrading, cutting, burning, applying strong chemicals, or transplanting friction ridge skin. Skin disease or side effects of drugs can also obliterate fingertips. *Distortion* comprises of cases of using plastic surgery to convert a normal friction ridge pattern into an unusual ridge pattern. Some portions of the skin are removed from the finger and grafted back onto a different position causing an unusual pattern. *Imitation* is when a surgical procedure is performed in such a way that the altered fingerprints appear as natural fingerprints, for example, by grafting skin from the other hand or a toe such that fingerprint ridge pattern is still preserved.

Cases of tampering with fingerprints to evade detection in criminal cases were reported as early as 1935. Cummins (1935) reported three cases of fingerprint alterations and presented images of before and after alterations. In recent years, border crossing applications have been targeted by altered fingerprint attacks. In 2009, it was reported that Japanese officials arrested an individual, who had paid a plastic surgeon to swap fingerprints between the right and left hands to evade detection (Heussner, 2009). Patches of skin from the individual's thumbs and index fingers were reportedly removed and then grafted onto the ends of fingers on the opposite hand. As a result, the individual's true identity was not detected when the person re-entered Japan illegally. In 2014, the FBI identified 412 records in its IAFIS, which are suspected to contain deliberate fingerprint

alterations (CJIS, 2015). In 2018, Business Insider reported that Eduardo Ravelo, who was added to the FBI's 10 Most Wanted list in October 2009, was believed to have had plastic surgery to alter his fingerprints to evade authorities (Weiss et al., 2020).

9.5 Presentation Attack Detection

The vulnerability of fingerprint systems to presentation attacks and the prominent media coverage of successful presentation attacks has created significant doubts in the minds of the public about the security of fingerprint systems. This is especially true in the case of fingerprint systems operating in an unsupervised scenario (e.g., authentication on a smartphone, secure facility access, self-check-in kiosks at airports), where the fingerprint presentation by a user is typically not monitored. Hence, the topic of presentation attack detection (PAD) has gained utmost importance over the past two decades (Marcel et al., 2019 and Sousedik and Busch, 2014). A series of fingerprint Liveness Detection (LivDet) competitions (Yambay et al., 2019) have been held since 2009 to benchmark various spoof detection solutions. Several large government-funded programs, including the U.S. IARPA ODIN program (2016) and European Union's TABULA RASA program (2013), were initiated with the goal of advancing the state-of-the-art in biometric (face, fingerprint, and iris) presentation attack detection. The world's largest biometric system with approximately 1.3 billion enrollments, India's Aadhaar program (2021), is also funding research to detect spoof fingers, faces, and irises.

In order to minimize the vulnerability of fingerprint recognition systems to presentation attacks, various PAD methods have been proposed. An ideal PAD method should satisfy the following requirements: (i) non-invasiveness, (ii) user-friendliness, (iii) low-cost, (iv) high efficiency, and (v) very low error rates. The PAD approaches proposed in the literature can be classified into *hardware* and *software*-based approaches.

9.5.1 Hardware-Based Approaches for Spoof Detection

The friction ridge skin is a layered tissue characterized by unique skin properties. A typical hardware-based PAD approach utilizes specialized sensor(s) to detect the signs of vitality (e.g., electrical properties, sub-surface imaging, blood pressure, pulse, and so on) to differentiate between bonafide fingers and PA instruments. Due to the additional hardware, these solutions are likely to be relatively expensive and have a larger form factor. Some of the hardware-based PAD approaches explored in the literature are:

- *Multi-spectral properties*: A specialized sensor can acquire multiple images of the finger surface as well as sub-surface characteristics of the finger using various wavelengths and polarization of light to separate bonafide fingers from PA instruments

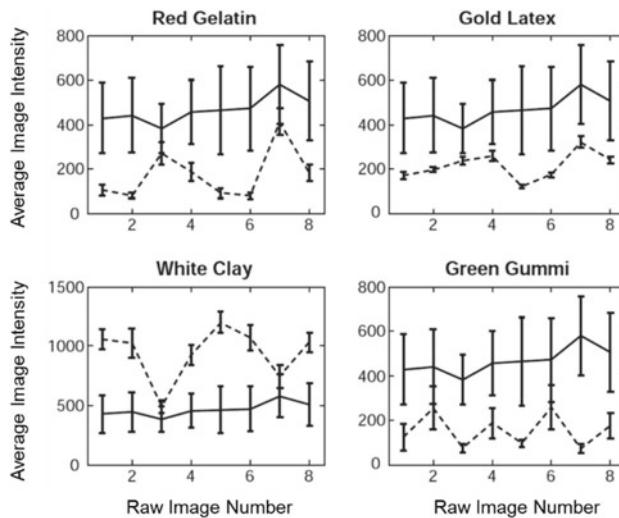


Fig. 9.12 Spectral differences between bona fide fingers and various fingerprint spoof types captured using a multi-spectral fingerprint scanner. The average image intensity for each of the eight direct-illumination images obtained by the scanner is plotted for bona fide fingers (solid lines, repeated in the four plots) and spoofs (dotted lines, different for each plot). Error bounds represent three standard deviations of the measured population for each sample class. It can be observed that all the selected spoof types are separable from bona fide fingers based on their average spectral properties. © Springer Nature. Reprinted, with permission, from Rowe et al. (2008)

(Rowe et al., 2008) (see Fig. 9.12). The optical properties of interest include absorption, reflection, scattering, and refraction properties under different lighting conditions (such as wavelength, polarization, coherence). However, it is not difficult to find a material (e.g., gelatin) whose optical properties are close to those of a live finger. A thin layer of material, like silicone, on top of a real finger, allows reproducing most of the optical properties of the real finger. Color that matches with the human tissue can be added to the synthetic material such as Play-Doh.

Another example of a multi-spectral sensor is the open-source *Build-It-Yourself* fingerprint reader, called *RaspiReader*, with a dual-camera design to provide two complementary streams of information, raw direct view RGB image and high contrast FTIR image (see Fig. 9.13) that are useful for spoof detection (Engelsma et al., 2019). Other approaches have also utilized short-wave infrared images to train deep neural networks to automatically learn features important for PAD (Tolosana et al., 2020).

- *Electrical properties:* The electrical conductivity of human tissue differs from the conductivity of many other synthetic materials such as silicone rubber and gelatin. The conductivity of the material presented to the fingerprint scanner can be measured to

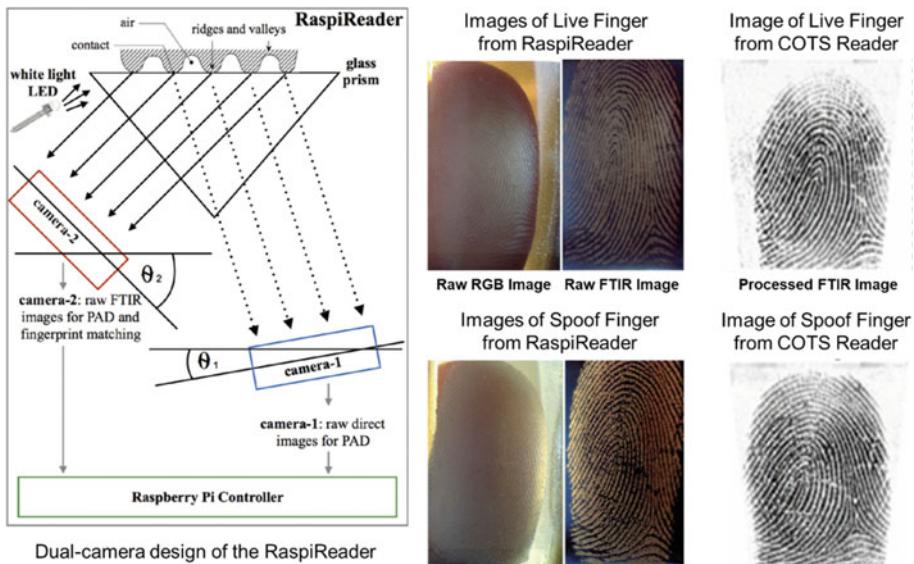


Fig. 9.13 Illustration of the open-source fingerprint reader called RaspiReader, which uses a dual-camera setup to provide two complementary streams of information useful for spoof detection (left column). While the raw images captured by the RaspiReader (middle column) are significantly different for live and spoof fingers, no such distinction can be observed in the images captured by a commercial-off-the-shelf (COTS) optical fingerprint reader (right column). © IEEE. Reprinted, with permission, from Engelsma et al. (2019)

differentiate a live finger from a fake finger. Shimamura et al. (2008) utilize impedance measurements module into a capacitive fingerprint sensor for PA detection. However, the conductivity of live fingers varies substantially depending on weather conditions such as humidity and temperature. If a fake finger is dipped in water, for example, its conductivity may be indistinguishable from that of a live finger. Relative Dielectric Constant (RDC) is also influenced by the humidity of the finger and thus will not be very effective in distinguishing a live finger from a fake. Moreover, simply applying alcohol on a fake finger changes its RDC significantly and thus can make it indistinguishable from live fingers. Another way to exploit the electrical properties of a human finger for PAD is to design a challenge-response mechanism. The sensor can be equipped with an electrode array to observe the response to the electric pulses transmitted into the fingertip during authentication (Yau et al., 2008).

- *Optical Coherence Tomography (OCT):* Since sub-surface characteristics are not directly visible and are not present in latent fingerprints, they cannot be obtained by an attacker (except in the case the finger owner colludes with the attacker). Therefore, presentation attack detection techniques based on these characteristics are inherently

strong against most of the presentation attacks. The Optical Coherence Tomography (OCT) technology (see Sect. 2.3.1) is capable of non-invasive in-depth imaging of human tissue with high resolution, providing crucial information of sub-surface morphology that can be utilized to detect fake fingerprints (Darlow et al., 2016). Unfortunately, today, OCT sensing devices are still bulky and expensive.

- *Ultrasound*: In-display fingerprint readers developed for smartphones by Qualcomm Inc. utilize acoustic response of ultrasonic waves reflected from the finger surface (see Sect. 2.3.5). Differences in the acoustic response of bona fide and fake fingers can be utilized for presentation attack detection (Agassy et al., 2019).
- *Odor*: Skin odor is different from odor of synthetic materials such as gelatin, latex, silicone, etc. Odor can be detected by using a “chemical sensor” such as those based on metal-oxide technology. Such sensors detect the odorants by detecting the tiny amounts of molecules that are evaporated from materials that have odor. An odor scanner (i.e., an electronic nose) contains an array of such odor sensors (Baldissera et al., 2006; Franco & Maltoni, 2007).
- *Biomedical properties*: These methods for finger vitality determination are based on measurements of pulse and blood pressure in the fingers. However, the pulse rate of the finger varies significantly from one person to another. In addition, pulse rate also varies depending on the physical activity and emotional state of the person at the time of acquisition. Furthermore, finger pressure on the sensor surface can change the pulse value quite dramatically and a single pulse measurement may take up to five seconds. Finally, if a wafer-thin silicone rubber is glued to a real finger, the heartbeat of the underlying finger will result in the detection of a pulse. Blood pressure and electrocardiogram sensors also have similar limitations.

9.5.2 Software-Based Approaches for Spoof Detection

Software-based approaches utilize either a static fingerprint image or a sequence of frames captured by the fingerprint reader to differentiate between bona fide fingers and PA fingerprints. These approaches are promising and more attractive as no additional hardware is required, and the PA detection capability can be upgraded over time by updating the software. Based on whether a single static image or a sequence of fingerprint images are utilized for PA detection, these methods are divided into two categories: *dynamic* and *static* methods (Marasco & Ross, 2015). One of the main PAD challenges is generalization across PA materials and sensors which were “unknown” at training time. Some software-based techniques designed to improve generalization are discussed in Sect. 9.5.4.



Fig. 9.14 Change in the moisture pattern of a live finger due to perspiration. Image courtesy of Stephanie Schuckers

Dynamic methods

- **Perspiration:** Bonafide fingers exhibit sweating over a period of time whereas PA instruments do not. In bonafide fingers, the perspiration phenomenon starts at the sweat pores and spreads along the ridge lines. The regions around the sweat pores progressively enlarge over time. To observe the sweating process, the finger needs to be placed on the scanner for a few seconds. Figure 9.14 shows fingerprint images of the same finger captured at three successive time instants. To quantify the temporal changes in the ridge characteristics caused by perspiration, the variations in pixel values between the first and the last image can be used as discriminative features. Tan and Schuckers (2006) analyzed the ridge signal (using multi-resolution texture analysis and wavelet analysis) extracted along the ridges (using a ridge mask) to detect the perspiration phenomenon from a single fingerprint image. A correct classification rate in the range of 84–100% was reported for three different fingerprint scanners. The limitations of perspiration-based methods stem from the varying amounts of moisture content in a finger and different finger pressure on the scanner surface.
- **Skin Distortion:** A skin distortion model can be learnt by observing the specific ways in which the finger skin distorts when a finger is pressed on a sensor surface. It is unlikely that fake fingers made of synthetic material fit the natural skin deformation model. In fact, skin is usually more elastic than most materials that are used to create replicas; furthermore, finger skin deforms in a specific way because it is anchored to the underlying derma and the deformation is influenced by the position and shape of the finger bone. Zhang et al. (2007) proposed computing distortion energies caused by different directional pressure for PA detection. The global distortion between a “normal” fingerprint and distorted fingerprint is modeled using a thin-plate spline (TPS) model between the set of paired minutiae; the distortion energy is computed based on the bending energy vector of the TPS model. Figure 9.15 depicts an illustration of this approach. Antonelli et al. (2006a, b) argue that to produce a relevant (measurable) distortion, the user could apply firm pressure on the scanner while simultaneously rotating the finger deliberately. Given several frames (at a frame rate of 10–15 frames

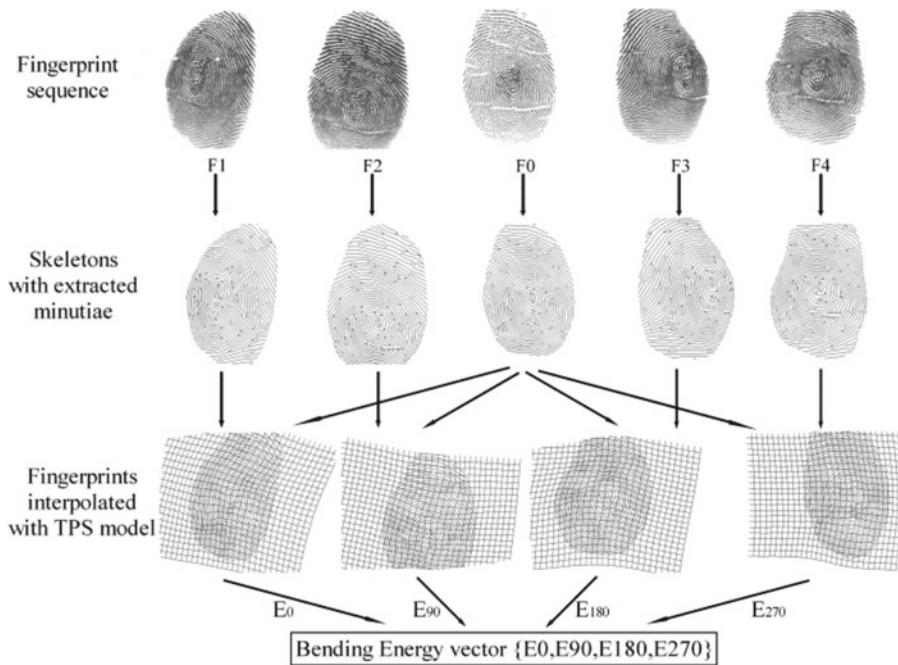


Fig. 9.15 Bending energy computation proposed by Zhang et al. (2007) to model the skin distortion. © Springer Nature. Reprinted, with permission, from Zhang et al. (2007)

per second), a feature vector from pairs of successive frames, known as DistortionCode was computed. User-specific DistortionCodes can be learnt during enrollment and compared with distortion measured at verification time.

- *Other dynamic features.* Rather than attempting to capture specific properties of human fingers such as perspiration or skin distortion, techniques have also been proposed to exploit the general temporal dynamics involved in the imaging of a fingerprint on a touch-based fingerprint sensor. Chugh and Jain (2020) proposed a deep neural network-based classifier applied on local fingerprint patches (centered around minutia points) extracted from 10 color frames for PA detection. Similarly, Plesh et al. (2019) used a fingerprint sensor with time-series and color-sensing capabilities and developed spatio-temporal dynamic features to facilitate the detection of presentation attacks.

Static approaches

Compared to dynamic methods which take a few seconds to acquire several images or a video sequence, static methods extract features from a single fingerprint impression and

are cheaper and faster. Static methods may extract handcrafted features, such as anatomical features, quality-based features, or/and textural features. These approaches utilize features that are either *hand-engineered* or *machine-learned*.

- *Hand-engineered Features*: Presentation attack detection approaches prior to 2015 primarily utilized hand-engineered features, including (i) anatomical features, (e.g., pore locations and their distribution), (ii) physiological features (e.g., perspiration), and (iii) texture-based features (e.g., power spectrum analysis and local descriptors). Marasco and Ross (2015) provide a comprehensive review of the various approaches employing hand-engineered features for presentation attack detection.
- *Machine-Learned Features*: Nogueira et al. (2016) explored the use of deep convolutional neural networks (CNNs) to automatically learn the features critical for presentation attack detection, and achieved state-of-the-art performance compared to previous approaches utilizing hand-crafted features. In order to avoid overfitting, they utilized transfer learning by fine-tuning CNN networks, originally pre-trained for object recognition. Pala and Bhanu (2017) employed a deep metric learning framework based on triplet loss to train a custom CNN architecture using randomly selected local patches. The randomness involved in the spoof fabrication process itself can generate some artifacts such as missing friction ridge regions, cracks due to dry skin, etc. The primary consequence of such noise is the creation of spurious minutiae in the fingerprint. The local regions around these spurious minutiae can provide salient cues to differentiate a PA fingerprint from a bonafide fingerprint. Chugh et al. (2018) utilized local patches extracted around fingerprint minutiae to train a CNN network. Zhang et al. (2019) utilized center of gravity-based local patches to train a custom CNN architecture using residual blocks and achieved the best performance in LivDet 2017 competition (Yambay et al., 2019).

9.5.3 Altered Fingerprint Detection

Detection of altered fingerprints is of high value to law enforcement and homeland security agencies to prevent known criminals (in the government watchlist) from evading the AFIS at border crossings and illegally entering the country. Existing approaches for detecting fingerprint alteration have primarily explored hand-crafted features to distinguish between altered and bonafide fingerprints. A bonafide fingerprint is expected to have a smooth orientation field except near the singular points (e.g., core and delta), and well spread and uniformly distributed minutiae points. However, an altered fingerprint typically exhibits discontinuous ridge flow even in non-singular regions, and densely located excessive spurious minutiae extracted along scars and obliterated regions.

Yoon et al. (2012) analyzed these orientation field and minutiae distribution-based characteristics to detect altered fingerprints. A polynomial model is utilized to estimate the orientation field, and a Parzen window method is used to estimate the minutiae density map. The errors between the observed and modelled orientation field and density maps are used to train a SVM model that outputs a “fingerprintness” score. A score close to 0 implies the input fingerprint image refers to an altered finger, and a score close to 1 denotes a bonafide fingerprint. Their method was tested on an operational database of 4,433 altered fingerprints from 270 subjects, resulting in 70.2% correctly identified altered fingerprints at a bonafide presentation classification error rate (proportion of bonafide fingerprints wrongly classified as altered fingerprints) of 2.1%. Ellingsgaard and Busch (2017) discuss methods for automatically detecting altered fingerprints based on analysis of two different local characteristics of a fingerprint image: identifying irregularities at the pixel-wise orientations and examining minutia orientations in local patches. Their method was tested on 116 altered and 180 unaltered fingerprint images collected from multiple sources. In comparison to hand-crafted features, Tabassi et al. (2018) employed a deep learning approach to automatically learn salient features which are crucial to detect and localize the altered regions. Their model was able to detect 99.24% of the altered fingerprints with a bonafide presentation classification error rate of 2.0% on an operational database of 4,815 altered fingerprints (from 270 subjects) and 4,815 bonafide images (see Fig. 9.16).

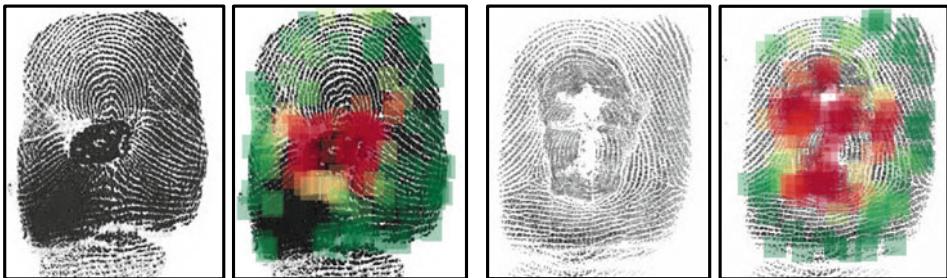


Fig. 9.16 Examples of altered fingerprint detection and localization. Local regions highlighted in red represent the altered portion of the fingerprint, whereas regions highlighted in green reflect the bonafide friction ridge area. © IEEE. Reprinted, with permission, from Tabassi et al. (2018)

9.5.4 PAD Performance Evaluation

Metrics

The performance of the PAD methods can be evaluated using the following metrics:

- *Attack Presentation Classification Error Rate* (APCER). It is the percentage of presentation attacks (spoof) fingerprints incorrectly accepted as bonafide (live) fingerprints.
- *Bonafide Presentation Classification Error Rate* (BPCER). It is the percentage of bonafide (live) fingerprints incorrectly rejected as presentation attack (spoof) fingerprints.

Both APCER and BPCER values depend on the threshold selected by the system operator. Hence, they are often reported as a pair, i.e., APCER at a specific value of BPCER or BPCER at a specific value of APCER. For instance, some studies report the APCER @ BPCER = 0.2%. This value represents the percentage of spoofs that can breach the fingerprint system security when the rate of legitimate users that are rejected is no more than 0.2%. Similarly, BPCER @ APCER = 1.0% represents the percentage of bonafide fingers that get rejected when the percentage of presentation attacks that go undetected is no more than 1.0%.

- *Detection Equal Error Rate* (D-EER). There is an operating point at which APCER becomes equal to BPCER and the error rate at this point is known as D-EER.
- *Average Classification Error Rate* (ACER). It is the average of APCER and BPCER at some fixed threshold and is computed as follows:

$$\text{ACER} = \frac{\text{APCER} + \text{BPCER}}{2}$$

If the number of presentation attacks and bonafide presentation attempts is equal, ACER represents the average error rate and (100-ACER) represents the average accuracy of the PAD algorithm. Since ACER and overall accuracy are dependent on the threshold, they are less useful compared to standard metrics such as APCER @ BPCER or D-EER.

Databases

The development and evaluation of robust PAD approaches require a large collection of bonafide and PA (spoof) fingerprints that are acquired by using a mix of PA materials, demographics, sensors, etc. Since 2001, some publicly available fingerprint PA databases have been utilized for algorithm benchmarking and to advance the state-of-the-art in PA detection. Some of the earlier databases were limited in size and contained a limited collection of PA materials and fingerprint sensors. Table 9.1 provides a summary of the recent, widely used, and publicly available databases. For more information on PA datasets collected prior to 2015, the reader is referred to Marasco and Ross (2015).

Table 9.1 Summary of publicly available datasets utilized to benchmark various fingerprint presentation attack detection (PAD) approaches

Database	Sensors (Technology)	#Bonafide/#PA images	#Subjects/ PA Materials
LivDet 2009	Biometrika (optical)	2,000/2,000	50/3
	CrossMatch (optical)	2,000/2,000	254/3
	Identix (optical)	1,500/1,500	160/3
LivDet 2011	Biometrika (optical)	2,000/2,000	50/5
	Digital Persona (optical)	2,000/2,000	100/5
	ItalData (optical)	2,000/2,000	50/5
	Sagem (optical)	2,000/2,000	56/5
LivDet 2013	Biometrika (optical)	2,000/2,000	75/5
	CrossMatch (optical)	2,500/2,500	235/4
	ItalData (optical)	2,000/2,000	250/5
	Swipe	2,500/2,500	75/4
LivDet 2015	Biometrika (optical)	2,000/2,500	51/6
	CrossMatch (optical)	3,010/2,921	51/5
	Digital Persona (optical)	2000/2,500	51/6
	Green Bit (optical)	2,000/2,500	51/6
LivDet 2017	Digital Persona (optical)	2,691/3,227	-/6
	Green Bit (optical)	2,700/3,240	-/6
	Orcanthus (thermal)	2,700/3,218	-/6
LivDet 2019	Digital Persona (optical)	2,019/2,224	-/7
	Green Bit (optical)	2,020/2,424	-/6
	Orcanthus (thermal)	1,990/2,288	-/6
MSU FPAD	CrossMatch (optical)	5,743/4,912	100/12
	Lumidigm (multispectral)	4,500/4,500	100/4

Generalization Performance

One of the major limitations of existing PAD methods is their poor generalization performance across “unknown” or novel PA materials that were not seen during the training of the PA detector. To generalize an algorithm’s effectiveness across spoof fabrication materials, called *cross-material* performance, some studies have approached spoof detection as an open-set recognition problem.¹ Table 9.2 presents a summary of the studies primarily focused on fingerprint PAD generalization. Rattani et al. (2015) applied the

¹ Open-set recognition problems address the possibility of encountering spoof classes during testing, which were not seen during training. Closed-set problems, on the other hand, evaluate only those spoof classes that the system was trained on.

Table 9.2 Summary of the studies primarily focused on fingerprint PAD generalization. In this table, ACER is the Average Classification Error Rate (ACER), D-EER is the Detection Equal Error Rate; APCER is the Attack Presentation Classification Error Rate, and BPCER is the Bonafide Presentation Classification Error Rate

Study	Approach	Database	Performance
Rattani et al. (2015)	Weibull-calibrated SVM	LivDet 2011	D-EER = 19.70%
Chugh et al. (2018)	MobileNet trained on minutiae-centered local patches	LivDet 2011–2015	ACER ≈ 0.97% (LivDet 2015), 2.93% (LivDet 2011, 2013)
Chugh and Jain (2019)	Identify a representative set of spoof materials to cover the deep feature space	MSU-FPAD v2.0	APCER = 24.76% @ BPCER = 0.2%
Engelsma and Jain (2019)	Ensemble of generative adversarial networks (GANs)	Custom database	APCER = 50.2% @ BPCER = 0.2%
Gonzalez-Soler et al. (2021)	Feature encoding of dense-SIFT features	LivDet 2011–2019	BPCER = 1.98% - 17% @ APCER = 1% (unknown PAI)
Tolosana et al. (2020)	Fusion of two CNN architectures trained on SWIR images	Custom database	D-EER = 1.35%
Zhang et al. (2019)	Slim-ResCNN + Center of Gravity patches	LivDet 2017	ACER ≈ 4.75%
Chugh and Jain (2021)	Style transfer between known spoof materials to improve generalizability against completely unknown materials	MSU-FPAD v2.0 and LivDet 2017	APCER = 8.22% @ BPCER = 0.2% (MSU-FPAD v2.0); ACER ≈ 4.12% (LivDet 2017)
Grosz et al. (2020)	Style transfer with a few samples of target sensor fingerprint images + Adversarial Representation Learning	LivDet 2015	APCER = 12.14% @ BPCER = 0.2% cross-sensor & cross-material

Weibull-calibrated SVM (WSVM), a variant of SVM based on properties of statistical extreme value theory, to detect spoofs made of new materials. Engelsma and Jain (2019) proposed using an ensemble of generative adversarial networks (GANs) on live fingerprint images with the hypotheses that features learned by a discriminator to distinguish between real live and synthesized live fingerprints can be used to separate live fingerprints from spoof fingerprints as well.

It has been shown that the selection of spoof materials used in training (known spoofs) directly impacts the performance against unknown spoofs. Chugh and Jain (2019) analyzed the material characteristics of 12 different spoof materials to identify a representative set of six materials that cover most of the spoof feature space. Although this approach can be used to identify if including a new spoof material in training dataset would be beneficial, it does not improve the generalization performance against materials that are unknown during training. Chugh and Jain (2021) proposed a style-transfer wrapper, Universal Material Generator (UMG), to improve the generalization performance of any spoof detector against novel spoof fabrication materials that were unknown during training of the spoof detector. It transfers the style (texture) characteristics between fingerprint images of known materials with the goal of synthesizing fingerprint images corresponding to unknown materials that may occupy the space between the known materials in the deep feature space.

Another dimension of generalization that is needed is with respect to fingerprint sensors. Fingerprint images captured using different fingerprint sensors, typically, have unique characteristics due to different sensing technologies, sensor noise, and varying resolution (see Fig. 9.17). This results in poor generalization performance in the *cross-sensor* scenario, where the spoof detector is trained on images captured using one sensor and tested on images captured from a different sensor. Grosz et al. (2020) utilize a few fingerprint samples from a target sensor to transfer its sensor texture characteristics to fingerprint images from a source sensor for domain adaptation. Additionally, they utilize adversarial representation learning to learn a sensor and material agnostic feature representation for improved generalization performance.

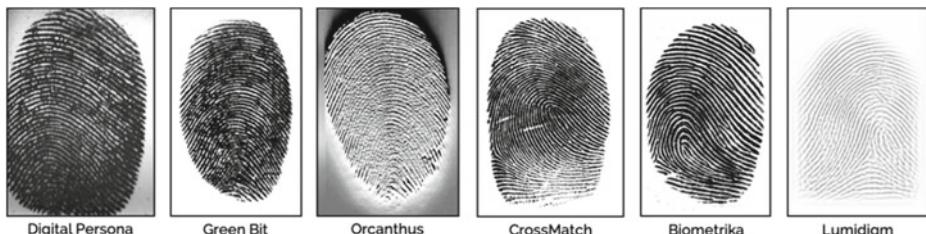


Fig. 9.17 Illustration of the differences in textural appearance of live scan fingerprints captured on six different fingerprint readers. © IEEE. Reprinted, with permission, from Grosz et al. (2020)

9.5.5 Challenges and Open Issues

Despite the significant improvement in the performance of fingerprint PAD algorithms over the last two decades, the results of the LivDet 2019 competition show that the best performing software-based PAD technique still has an average classification error rate (ACER) of approximately 3.83% for detecting known presentation attacks. It has been shown that the PAD error rates of software-based approaches suffer up to a three-fold increase when applied to datasets containing PA materials not seen during training, denoted as *cross-material* generalization (Chugh & Jain, 2021). A similar performance gap exists for *cross-sensor* generalization, in which presentation attack algorithms are applied to fingerprint images captured on new fingerprint sensor devices that were not seen during training. Furthermore, fingerprint PAD systems are not designed to work in isolation and need to be integrated with the overall fingerprint recognition pipeline. Errors made by the PAD algorithm will impact the accuracy of the fingerprint recognition system. For example, an integrated testing protocol followed in LivDet 2019 showed that the best fingerprint system with PAD had an overall accuracy of only 96.88% (includes errors made by both the PAD algorithm and the fingerprint matcher), which is typically low for a system working on good quality live-scan fingerprint images.

Apart from the problem of poor generalization, there are other challenges and open issues in this domain:

- *Interpretability*: The use of convolutional neural networks (CNNs) has revolutionized the research in fingerprint PAD, achieving unprecedented performance improvements. But such solutions are usually considered as “black boxes” shedding little light on how and why they achieve such a high performance. It is critical to gain insights into what CNNs learn to differentiate between bonafide fingers and PA instruments. Some of the techniques currently explored are through visual exploration, i.e., to identify the image regions that are responsible for the final predictions (see Fig. 9.18).
- *Efficiency*: The emergence of fingerprint applications working on smartphones and embedded devices has brought forth the need to design computationally efficient PAD approaches for low-resource environments.
- *Adaptive Adversary*: Developments in cryptographic systems have demonstrated that any security solution based on the secrecy of an algorithm (security through obscurity) does not provide satisfactory results over a long period of time. This is because the secret needs to be broken only by a single person and once this happens (it eventually always does), the entire solution immediately falls apart (such attempts are often posted on the Internet). Therefore, we should assume that the PAD approach being used by the fingerprint system is available in the public domain. Based on this knowledge, it may be easy for an adversary to design a PA instrument that will circumvent a specific PAD algorithm. For example, if it is known that a hardware-based PAD approach measures the pulse to check finger vitality, one could design a three-dimensional mold of a finger

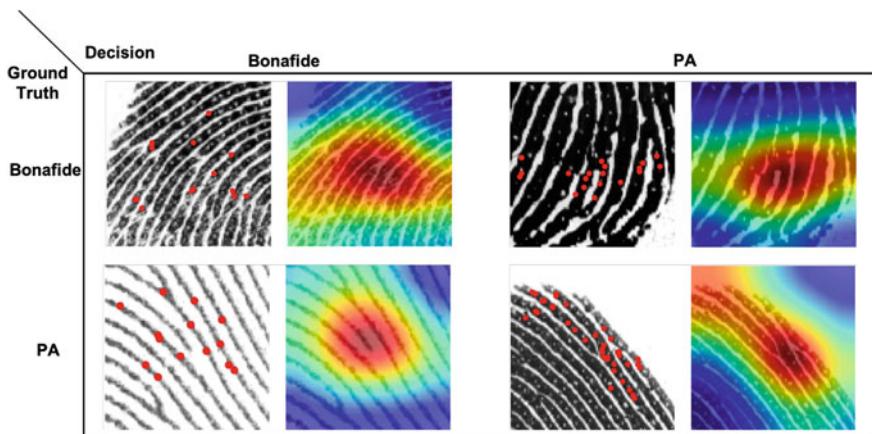


Fig. 9.18 Salient fingerprint patches that are responsible for the prediction made by a convolutional neural network (CNN)-based fingerprint PAD. These regions are identified using a visual exploration technique called CNN-Fixations (Mopuri et al., 2018). Image courtesy of Tarang Chung (Chugh, 2020)

that has a friction ridge pattern on its outer surface and a pulse generating device inside it. Some characteristics may be easier to simulate (such as thermal or optical property of human skin) than others (such as the sweating process, or the sub-surface features). Similarly, it is well-known that CNNs are vulnerable to cleverly crafted adversarial samples, which may also be true for CNN-based PAD methods.

9.6 Template Protection

In Sect. 9.3.5, we have already seen why typical cryptographic approaches (e.g., hashing and encryption) used in most knowledge-based authentication systems are not appropriate for fingerprint systems. This has led to the development of a vast array of specialized techniques for securing biometric data, including fingerprints. While template protection is a challenging problem in any biometric system (e.g., fingerprint, face, and iris), the challenge is more pronounced in fingerprint systems due to the large intra-class variations in multiple impressions of the same finger. Unlike iris recognition, where a fixed-length binary string representation called the iriscode is the de facto standard, fingerprint systems often use minutiae-based unordered set representations that are inherently more difficult to secure.

The general framework for biometric template protection techniques is shown in Fig. 9.19. Rather than storing the biometric template in its original or native form, a finger-

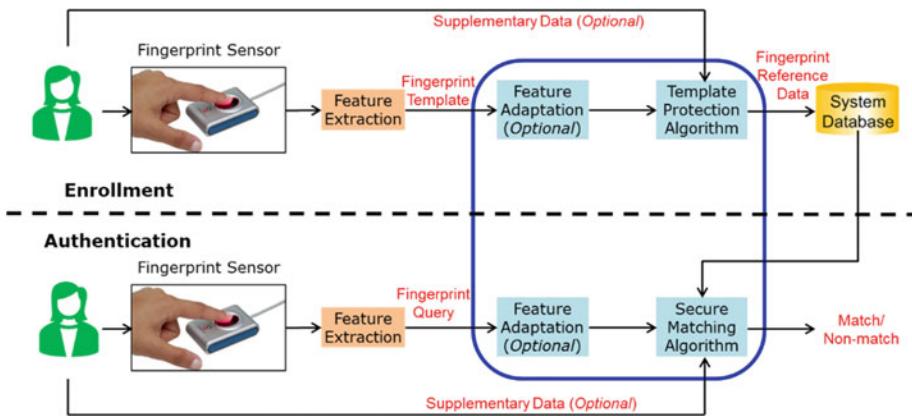


Fig. 9.19 General framework of a fingerprint system with template protection. © IEEE. Reprinted, with permission, from Nandakumar and Jain (2015)

print template protection algorithm generates and stores a *protected fingerprint reference* derived from the original template. Note that the term protected fingerprint reference not only includes the protected fingerprint information (e.g., minutiae), but also other system parameters or values (e.g., cryptographic hashes) that need to be stored, as well as any side information (e.g., information required for fingerprint alignment, quality of the features, etc.) that directly does not leak information about the user identity. On the other hand, *supplementary data* refers to entities that are not stored in the database but are required during both enrollment and authentication. Examples of supplementary data include a password or secret key provided by the user in addition to his fingerprint. The use of supplementary data is optional, but if used, it provides an additional factor of authentication.

In the template protection framework shown in Fig. 9.19, an optional step called feature adaptation has been introduced. Though this module does not play any direct role in protecting the fingerprint template, its objective is to minimize intra-subject variations (e.g., through quantization) in the fingerprint features to a level that can be handled by the secure matching algorithm. In many applications, feature adaptation module also represents the original features in a simplified form (e.g., a binary string) without diluting their distinctiveness. A detailed discussion of various feature adaptation strategies is presented in Sect. 9.6.5.

9.6.1 Desired Characteristics

In the context of biometric template security, the protected biometric reference is typically considered as public information that is available to any adversary. Hence, the protected biometric reference should satisfy the following three properties:

1. *Non-reversibility* or *Irreversibility*: It should be computationally hard² to obtain the original fingerprint template from an individual's protected fingerprint reference. This property prevents the abuse of stored fingerprint data for launching presentation attacks, thereby improving the security of the fingerprint system. One of the consequences of this requirement is that the fingerprint matching needs to be performed in the transformed space of the protected reference, which may be very difficult to achieve with high accuracy.
2. *Revocability* or *Renewability*: It should be computationally hard to obtain the original fingerprint template from multiple instances of protected fingerprint reference derived from the same fingerprint of an individual. Furthermore, it should be possible to produce a very large number of protected references (to be used in different applications) from the same fingerprint template. This makes it possible to revoke and re-issue new instances of protected fingerprint reference when an enrollment database is compromised. Moreover, this prevents an adversary from obtaining the original template by compromising multiple fingerprint databases where the same individual may be enrolled.
3. *Non-linkability* or *Unlinkability*: It should be computationally hard to ascertain whether two or more instances of protected fingerprint reference were derived from the same fingerprint of a user. The non-linkability property prevents cross-matching across different applications, thereby avoiding function creep and preserving the privacy of the individual.

Apart from satisfying the above three properties, an ideal template protection algorithm must not substantially degrade the recognition performance (accuracy) of the fingerprint system (or at least the degradation must happen smoothly). In many applications of fingerprint identification, especially those involving millions of enrolled identities (e.g., border crossing and national ID systems), accuracy is of paramount importance. If the accuracy degrades substantially, it will constitute the weakest link in the security equation. For example, instead of reversing the enrollment template, the adversary may attempt a brute-force attack with a dictionary of fingerprints to cause a false accept. Moreover, issues such as throughput (number of fingerprint comparisons that can be performed in unit time) and template size must also be considered in real-world applications.

² A problem is considered to be computationally hard or difficult if it cannot be solved using a polynomial-time algorithm.

Note that all the above properties must be satisfied simultaneously for a template protection scheme to be effective in practice. For example, if a protected fingerprint reference is reversible but not unlinkable, the adversary would be able to carry out function creep, even though the fingerprint data is protected. Designing methods to satisfy all the above-mentioned properties is a very challenging task. In particular, template protection techniques typically result in an unacceptable trade-off between irreversibility and matching accuracy (Nagar, Nandakumar, and Jain (2010) and Wang et al. 2012) due to the following reason. Maximizing irreversibility implies that the protected fingerprint reference should leak as little information about the original template as possible. However, high matching accuracy can be achieved only when the protected biometric reference retains all the discriminatory information contained in the original template. Overcoming this conundrum is critical to the development of effective template protection schemes.

The first step towards solving the problem of accuracy versus security trade-off is to clearly define the notion of security, establish metrics to quantify security properties such as irreversibility and unlinkability and develop methodologies to compute such metrics. Often, the false match rate (FMR) of a fingerprint system is considered as an upper bound on the irreversibility of a protected fingerprint reference. Since most practical fingerprint systems restrict the number of failed authentication attempts, it is usually not possible to mount online zero effort attacks. Therefore, it may be better to consider vulnerability to zero effort attacks as a distinct threat and report the FMR of the fingerprint system before and after the application of fingerprint template protection. Ideally, the FMR should be included as part of the recognition performance and not security analysis. Furthermore, the FMR of the biometric system after template protection should be reported based on the assumption that the attacker has full knowledge about the system, including access to any supplementary data (if used).

Since irreversibility denotes the difficulty in obtaining (either exactly or within a small margin of error) the original fingerprint template from an individual's protected fingerprint reference, a direct measure of irreversibility is the conditional Shannon entropy of the original fingerprint template \mathbf{x} given the protected biometric reference \mathbf{v} , i.e., $H(\mathbf{x}|\mathbf{v})$. The entropy of a random variable is the average level of information or uncertainty in the variable. Therefore, $H(\mathbf{x}|\mathbf{v})$ measures the average uncertainty in estimating \mathbf{x} given the knowledge of \mathbf{v} . Note that $H(\mathbf{x}|\mathbf{v}) = H(\mathbf{x}) - I(\mathbf{x};\mathbf{v})$, where $H(\mathbf{x})$ is the entropy of the original fingerprint template \mathbf{x} and $I(\mathbf{x};\mathbf{v})$ is the mutual information between \mathbf{x} and \mathbf{v} . Sometimes, $I(\mathbf{x};\mathbf{v})$ is also referred to as *entropy loss*, which measures the amount of information leaked by the protected biometric reference about the biometric template. Entropy loss is a useful measure to compare multiple template protection schemes applied to the same fingerprint data. In this scenario, since $H(\mathbf{x})$ is constant, the scheme with a lower entropy loss should be preferred because it will lead to larger $H(\mathbf{x}|\mathbf{v})$. While the conditional Shannon entropy is a good measure of the average difficulty in inverting a protected biometric reference, researchers have also proposed the use of *min-entropy*

(Dodis et al., 2008) to account for the worst-case scenario. The min-entropy measures the uncertainty in predicting the most likely value of a discrete random variable.

While the above metrics for measuring irreversibility are theoretically sound, they are not easy to compute for an arbitrary fingerprint template protection scheme. In most biometric cryptosystems, the inherent properties of the underlying error correction technique can be used to establish upper bounds on the entropy loss (Dodis et al., 2008, Ignatenko & Willems, 2009, 2010; Lai et al., 2011). Typically, the entropy loss is an increasing function of the error correction capability of the system. In other words, if larger tolerance for intra-subject variations is desired, the entropy loss will be higher. Consequently, the resulting protected fingerprint references will leak more information about the original template. Since the above bounds are usually derived based on simplifying assumptions about the fingerprint feature distribution, their utility will depend on the extent to which the given fingerprint features conform to these assumptions. Even when a reliable estimate for the entropy loss is available, it is still difficult to directly compute $H(\mathbf{x}|\mathbf{v})$. This is because of the complexity in estimating the entropy of fingerprint features ($H(\mathbf{x})$). The primary difficulty in estimating the entropy of fingerprint features is the lack of statistical models to accurately characterize the intra- and inter-subject variations. This is closely related to the individuality of fingerprints discussed in Chap. 8.

9.6.2 Template Protection Approaches

Numerous template protection techniques have been proposed in the literature with the objective of ensuring non-invertibility, revocability, and non-linkability without compromising on the recognition performance. The ISO/IEC 24745 (2011) standard on Biometric Information Protection provides a general guidance for the protection of biometric information. According to this standard, a protected biometric reference is typically divided into two parts, namely, *pseudonymous identifier* (PI) and *auxiliary data* (AD). Depending on how these two components (PI and AD) are generated, fingerprint template protection schemes can be broadly categorized as: (i) feature transformation approach and (ii) biometric cryptosystems (Nandakumar & Jain, 2015).

- *Feature transformation:* In the feature transformation approach (see Fig. 9.20), a noninvertible or one-way function is applied to the fingerprint template. While the transformed template is stored in the database as PI, the transformation parameters are stored as AD. During authentication, the AD makes it possible to apply the same transformation function to the query fingerprint and construct PI', which is compared to the stored PI. Thus, fingerprint matching takes place directly in the transformed domain. While some feature transformation schemes satisfy the irreversibility property only when the supplementary data (e.g., key or password) is assumed to be a secret, there are other techniques that can generate irreversible templates without the need for

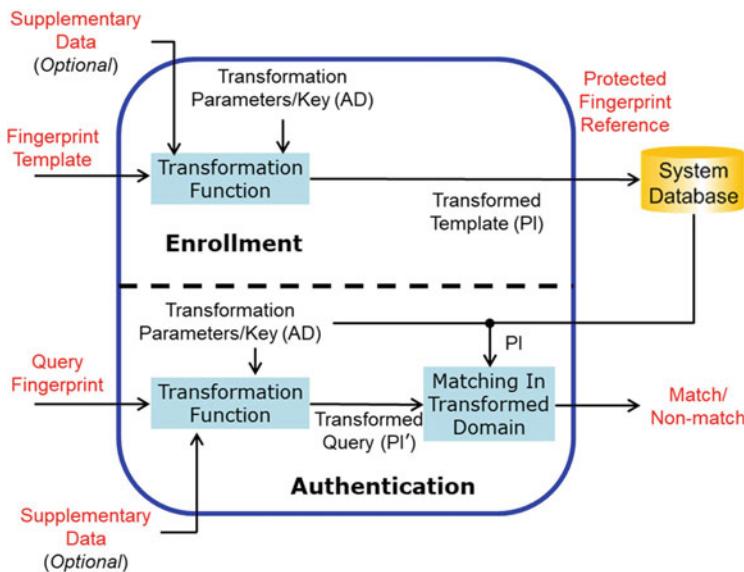


Fig. 9.20 Typical workflow of the feature transformation approach for fingerprint template protection. © IEEE. Reprinted, with permission, from Nandakumar and Jain (2015)

any secrets. The former group of algorithms are, by definition, multi-factor authentication schemes, which are feasible only in certain access control applications. The schemes belonging to the latter category are useful in applications (e.g., law enforcement), where it may not be feasible or desirable to allow user-specific supplementary data. Feature transformation approaches are discussed in Sect. 9.6.3.

- **Biometric cryptosystem:** In biometric cryptosystems, the auxiliary data is often referred to as a secure sketch (see Fig. 9.21), which is typically derived using error correction coding techniques. While the secure sketch in itself is insufficient to reconstruct the original fingerprint template (irreversibility), it does contain adequate information to recover the original template in the presence of another fingerprint impression that closely matches with the enrollment sample. The secure sketch is either obtained as the syndrome of an error correction code applied to the fingerprint template (*key generating cryptosystem*) or by binding the fingerprint template with an error correction codeword that is indexed by a cryptographic key. Note that this cryptographic key is independent of the enrollment template and hence, this approach is known as *key-binding cryptosystem*. It must be emphasized that a key-binding fingerprint cryptosystem is not the same as encrypting the fingerprint template using a key based on standard encryption techniques. Unlike an encrypted template, the secure sketch embeds both the fingerprint template and cryptographic key within a single entity and reveals minimal information about them individually unless presented with a matching fingerprint. A cryptographic

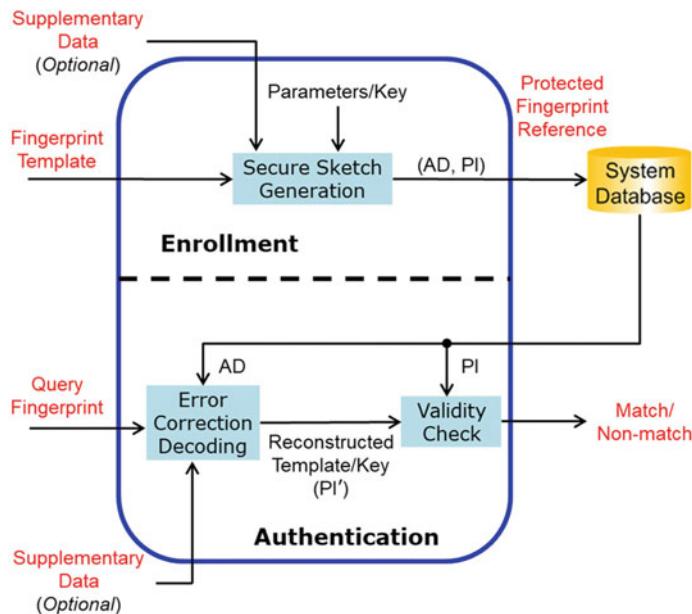


Fig. 9.21 Typical workflow of a fingerprint cryptosystem used for fingerprint template protection.
© IEEE. Reprinted, with permission, from Nandakumar and Jain (2015)

hash of the original fingerprint template or the key used to index the error correction codeword is stored as PI. Matching in a biometric cryptosystem is performed indirectly by attempting to recover the original template using the secure sketch (AD) in conjunction with the query fingerprint features. The recovered template is used to regenerate a new pseudonymous identifier (PI'), which is compared to the stored PI to determine whether the template and query match.

Both the template protection approaches have their own strengths and limitations. Revocability is easier to achieve in the feature transformation approach, which is why these schemes are also loosely referred to as *cancelable biometrics* (Patel et al., 2015). But the challenge in the feature transformation approach is finding an appropriate transformation function that provides strong irreversibility guarantees, but at the same time tolerant to intra-subject variations. In the case of feature transformation, it is often difficult to theoretically measure the entropy loss introduced by the transformation scheme. Consequently, the irreversibility of feature transformation schemes is typically measured empirically based on the computational complexity of the best-known template inversion attack.

The strength of biometric cryptosystems is the availability of bounds on the information leaked by the secure sketch (entropy loss) if we assume that the biometric data distribution is known (Dodis et al., 2008; Ignatenko & Willem, 2009). On the flip side,

most biometric cryptosystems require the features to be represented in standardized data formats like binary strings and point sets, which often leads to loss of discriminatory information and consequent degradation in recognition accuracy. Moreover, most biometric cryptosystems use linear error correction codes, where any linear combination of codewords is also a codeword. Consequently, if two secure sketches are derived from the same fingerprint data using different codewords, a suitable linear combination of these two sketches is highly likely to result in a decodable codeword. This paves the way for verifying whether the two secure sketches are derived from the same finger, thereby making them linkable. Thus, it is difficult to achieve non-linkability in fingerprint cryptosystems. One way to overcome the above limitations is to apply a feature transformation function to the fingerprint template before it is protected using a fingerprint cryptosystem. Since this involves both feature transformation and secure sketch generation, such systems are known as hybrid biometric cryptosystems (Boult et al., 2007; Feng et al., 2010).

Biometric cryptosystems also have an interesting additional benefit. It is well-known that key management is one of the thorny issues in most cryptographic systems. This is because an encrypted secret is secure only as long as the decryption key is secure (see Fig. 9.22a). The hard problem of cryptographic key management can be alleviated by using a biometric/fingerprint cryptosystem, where the key generated as part of the secure matching process can be used in another application to decrypt an encrypted secret (see

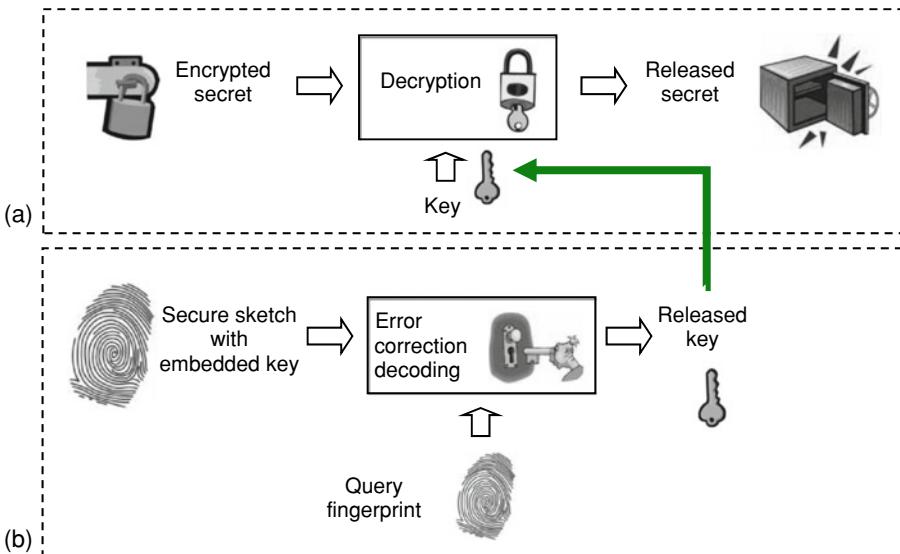


Fig. 9.22 Leveraging a fingerprint cryptosystem to alleviate the key management problem in cryptographic systems. **a** Typical workflow of a cryptographic system and **b** illustration of how a fingerprint cryptosystem can be used as a secure key release mechanism

Fig. 9.22b). Thus, it can be ensured that an authorized user is able to access an encrypted secret only if there is a successful biometric authentication.

Apart from feature transformation and biometric cryptosystems, another promising approach is a secure computation based on *homomorphic encryption* (Acar et al., 2018). Homomorphic encryption (HE) provides the ability to directly perform certain mathematical operations on the encrypted data. For example, if the fingerprint template is encrypted using a HE scheme, it is possible to directly perform matching in the encrypted domain, without decrypting the template. In this sense, homomorphic encryption is conceptually similar to feature transformation, but it provides very strong cryptographic guarantees on the security of the encrypted templates. In particular, *fully homomorphic encryption* (FHE) schemes have been developed by the cryptographic community over the past decade (Gentry, 2009) and these FHE constructs allow both addition and multiplication operations on encrypted data. Consequently, it is now feasible to compute any polynomial function of the encrypted data, thereby enabling more complex fingerprint matching algorithms to be implemented in the encrypted domain. While the FHE approach offers the attractive proposition of performing biometric matching directly in the encrypted domain, it typically comes at the cost of a significant increase in the computational burden and communication overhead (Bringer et al., 2013). Despite this limitation, Engelsma et al. (2021) have shown that with compact fingerprint feature representations, it is possible to perform matching in the encrypted domain with a matching time of 1.26 ms, which is perfectly acceptable in most authentication applications. Another minor limitation of encrypted domain matching is that the result of the matching process (say, match score) remains encrypted until it is decrypted using the private key. This requires careful design of the authentication system such that the matcher and decision modules are not simultaneously compromised. While the database and matcher operate completely on encrypted data, the decision module holds the key to decrypt the matching result and make a match/non-match decision.

9.6.3 Feature Transformation

Conceptually, a fingerprint template can be protected by transforming the unprotected template into another representation space through a *non-invertible transform*. The most popular non-invertible transform is a one-way cryptographic hash function, denoted as $c = \text{Hash}(x)$, which is used together with a verification function $V(x,c) \Rightarrow \{\text{True}, \text{False}\}$. Ideally, $V(x,c) \Rightarrow \text{True}$ only if $c = \text{Hash}(x)$. In addition, the pair of functions must have the following properties:

- *Collision resistance*: It should be hard to find x and y such that $\text{Hash}(x) = \text{Hash}(y)$ if $x \neq y$.

- *Pre-image resistance*: If an adversary has access to a hashed code $c = \text{Hash}(x)$ and knows the hash function $\text{Hash}(\cdot)$, the only way to determine data x^* such that $V(x^*, c) \Rightarrow \text{True}$ is to exhaustively search over x (brute force attack).

Thus, the security (cryptographic strength) provided by the one-way hash function is dependent on the information content of the data x . Hashing techniques are extensively used in password-based authentication systems; passwords are hashed and stored in the database during user enrollment (see Fig. 9.23a). When an input password is entered, it is also hashed and compared with the stored hashed password. Since the transformation is irreversible in the cryptographic sense, the original password cannot be recovered even if the exact transformations, as well as the transformed password, are known. A different transform (or a differently parameterized transform) is used for a different application thus avoiding cross-use of the passwords.

The same concept could in principle be applied to fingerprints. Instead of maintaining a database of fingerprint templates, the hashes of the templates can be stored; at each recognition attempt, the query feature set is also hashed, and the fingerprint comparison is performed in the transformed space. Revocability can be achieved by re-enrolling the same user applying a different transform (or different parameters of the same transformation function) to his fingerprints. Irreversibility is guaranteed by the construction of the hash function itself. So, this approach is very attractive in theory.

However, there is a significant difference between a password and fingerprint hashing. Passwords are identical during different authentication attempts, but fingerprint images at different verification attempts are never identical, and this prevents the same hash to be obtained. A major obstacle in comparing hashed fingerprint templates is recovering the correct alignment between the two enrolled and query fingerprints. Methods to overcome this challenge are discussed in Sect. 9.6.5. Even if the fingerprints are pre-aligned, a robust hashing technique is required and matching in the transformed domain needs to be invariant to intra-class variation (see Fig. 9.23b). Therefore, it is very difficult to find non-invertible feature transforms that are both cryptographically secure and accurate from a biometric point of view.

Ratha et al. (2001) pioneered the concept of feature transformation for template protection. In Ratha et al. (2007), the authors extend their conceptual work by providing three specific non-invertible transforms for fingerprints. These transformation functions were used to transform fingerprint minutiae data such that existing minutiae matchers can still be applied to match the transformed minutiae. The key conclusion of this study is that transformation functions need to be locally smooth to preserve matching accuracy. On the other hand, if the transform is globally smooth, it would be easy to invert it, and hence it cannot be cryptographically secure. The challenge lies in finding a reasonable balance between these two competing requirements. As a result, the authors recommend the use of a functional surface folding transform that is locally smooth but is not globally smooth (Fig. 9.24); the function has “folds”, that is, multiple locations in the original space are

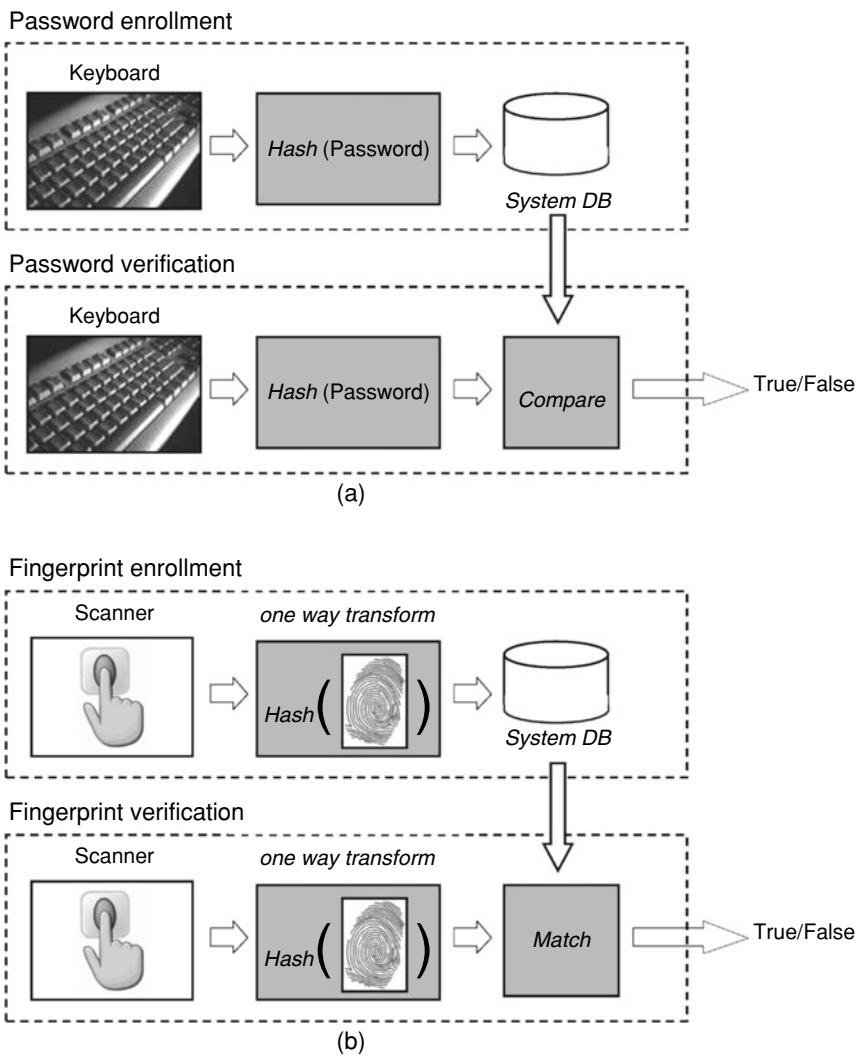


Fig. 9.23 Hashing approach for fingerprint template protection. **a** Passwords are typically stored in the database after they are hashed; when a new password is received, it is hashed and compared with the hashed enrollment. If a person has access to the database of hashed passwords, a password is not compromised. In **b**, a similar analogy is applied to fingerprint recognition. Only a robust hash of the original fingerprint template is stored and thus, if an adversary has access to the database, the fingerprint information is not compromised

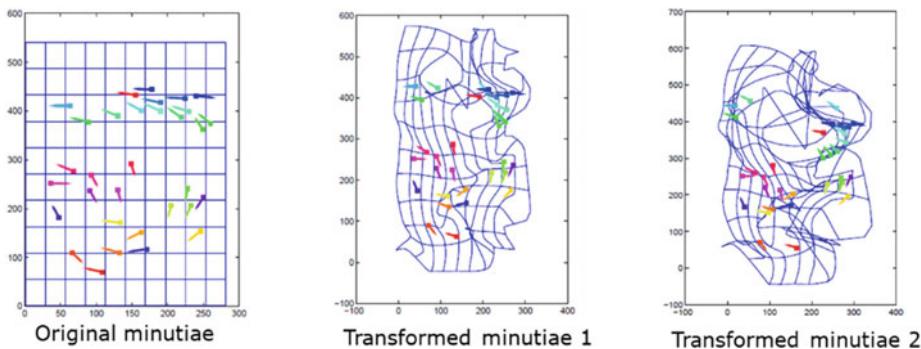


Fig. 9.24 In the feature transformation function proposed by Ratha et al. (2007), both the position and the orientation of the minutiae are changed by a surface folding function. Conceptually, the minutiae are embedded in a sheet, which is then crumpled. This function is locally smooth, but not globally smooth. As the transformation parameter is increased (as we move from the middle image to the right image), the irreversibility increases, but the matching accuracy decreases. This demonstrates the non-invertibility versus accuracy tradeoff in this approach

mapped to the same location in the transformed space. Conceptually, this is the same property that gives non-invertibility to standard cryptographic hash functions. However, the proposed transform has a low “degree of folding”, with only 8% of minutiae having their neighbors perturbed after this transformation. As a result, despite the high matching accuracy, the irreversibility provided by the proposed approach is not strong.

Other interesting techniques in this category include Sutcu et al. (2007b), Tulyakov et al. (2007), Ferrara et al. (2012), and Moujahdi et al. (2014). However, none of these techniques convincingly strike the right balance between the various requirements such as matching accuracy, irreversibility, revocability, and unlinkability.

If supplementary data is introduced into the feature transformation framework, it becomes possible to achieve very high matching accuracy, revocability, as well as unlinkability. The major drawback is that irreversibility now entirely depends on maintaining the secrecy of the supplementary data. Consequently, these techniques inherently fall under the category of multi-factor authentication. *Biohashing* (Teoh et al., 2006) is the most well-known template protection algorithm that comes under this approach. Once the supplementary data falls into the hands of an adversary (often referred to as the *stolen key scenario*), the “randomness” fades out (Kong et al., 2006) and the performance of these techniques typically drops below that of other non-invertible transformation schemes described earlier. In the stolen key scenario, even if a small amount of information is lost due to quantization, a fairly good approximation of the original fingerprint template can be recovered from the protected reference (Jain et al., 2008). To improve the security of

biohashing approaches, it is recommended that the key is not stored but rather remembered by the user, but this reintroduces the weaknesses of password-based authentication schemes that we are trying to overcome.

9.6.4 Fingerprint Cryptosystems

The idea of fingerprint cryptosystems was conceptualized by the cryptographic community as part of their search for techniques to extract cryptographic keys from noisy input data. Since biometric data are inherently noisy (e.g., different impressions of the same finger are similar, but not identical) and can be easily acquired from the user at the time of key generation, they are considered as suitable inputs for the key generation process. However, reliable cryptographic key generation is possible only if the noise in the data can be eliminated and the standard tool used for this purpose in the field of information theory is error correction coding. Thus, almost all biometric cryptosystems use some variant of error correction coding schemes. Dodis et al. (2008) proposed two constructs to turn biometric data into keys usable for general purpose cryptographic application: (i) *secure sketch* and (ii) *fuzzy extractor*.

- The secure sketch addresses the intra-class variability in the fingerprint. A protected fingerprint reference generated through this approach is called a “sketch of the unprotected fingerprint template”. The sketch leaks only minimal information about the original template and thus can be made public. The original fingerprint template can be exactly reconstructed from the secure sketch given a feature set that is sufficiently close to the original unprotected template.
- The fuzzy extractor goes beyond the concept of secure sketch and addresses both the intra-class variability in the fingerprint, as well as nonuniformity, that is, it extracts a uniform random string (key) from its input in an error-tolerant way. If the fingerprint input changes somewhat, the extracted key remains the same. In the context of fingerprint template protection, the secure sketch construct is more important than a fuzzy extractor.

In a key-binding biometric cryptosystem, a cryptographic key and the given fingerprint template are monolithically bound together within a cryptographic framework to generate the secure sketch. Juels and Wattenberg (1999) proposed a framework called *fuzzy commitment*. The user chooses a random codeword C of an error correcting code. Then the hash of C (i.e., $\text{Hash}(C)$) is stored as the *pseudonymous identifier* (PI), and the difference between the original template T and C , is also stored as *auxiliary data* (i.e., $AD = T - C$). This auxiliary data or difference vector ($T - C$) binds the codeword C to the template T . At verification time, the feature set denoted by I , is used to compute the vector $C' = I - (T - C)$; if I is similar to T , C' is expected to be similar to C . Error correction is then

applied to C' to get C'' . Finally, the stored $\text{Hash}(C)$ is compared to $\text{Hash}(C'')$; the comparison is successful if I is sufficiently close to T such that the correct codeword can be exactly recovered (i.e., $C'' = C$). Concrete implementations or experimental results were, however, not reported in Juels and Wattenberg (1999). The fuzzy commitment scheme requires the fingerprint representation to be alignment-free and ordered (i.e., there should exist a way to order single features within the feature set). Since most error correction schemes work with binary data, we also need to represent the fingerprint as a binary string.

Juels and Sudan's (2002) *fuzzy vault* method can be thought of as an order-invariant version of the Juels and Wattenberg (1999) method. In other words, the fuzzy vault method does not require the biometric features to be an ordered list of elements. This bodes very well for fingerprint recognition since the most popular fingerprint features (i.e., the minutiae) are not characterized by a natural ordering, and any attempt to sort them could lead to robustness issues. In the fuzzy vault method, a user, say Alice, places a secret value K (e.g., her private cryptographic key) in a vault and locks (secures) the vault by using an unordered set T_A (e.g., a list of minutiae points extracted from her fingerprint). Another user, say Bob, using another unordered set I_B , cannot unlock the vault (and thus cannot access the secret K) unless I_B is similar enough to T_A . To construct the vault, Alice performs the following operations:

- Selects a polynomial p that encodes K (e.g., by fixing the coefficients of p according to K).
- Computes the polynomial projections, $p(T)$, for the elements of T .
- Adds some noise (i.e., randomly generated chaff points having projection values different from that corresponding to p) to derive the final point set V (that corresponds to the auxiliary data for the fuzzy vault approach).

When Bob tries to learn K (e.g., by finding p), he uses his own unordered set I_B . If I_B is not sufficiently similar to T (which is expected since Bob's fingerprint will be very different from Alice's) he will be unable to locate many points in V that lie on p , especially in light of the chaff points that will mislead Bob's efforts. Thus, Bob will not be able to obtain K . On the other hand, when Alice needs to retrieve K from the vault, she will provide a new unordered set I_A derived from a different impression of her finger. Now, since I_A is expected to be sufficiently similar to T_A , by using error correcting codes (e.g., Reed Solomon codes), Alice will be able to reconstruct p , and hence her secret key K . See Fig. 9.25 for examples of the enrollment and the verification processes using the fuzzy vault technique.

Numerous researchers have attempted to apply the secure sketch construct for fingerprint template protection. Most of these methods differ in the following aspects: (a) how the fingerprints are pre-aligned or alignment-free features are extracted, (b) how to adapt the native fingerprint representation into a format (e.g., unordered sets with set difference

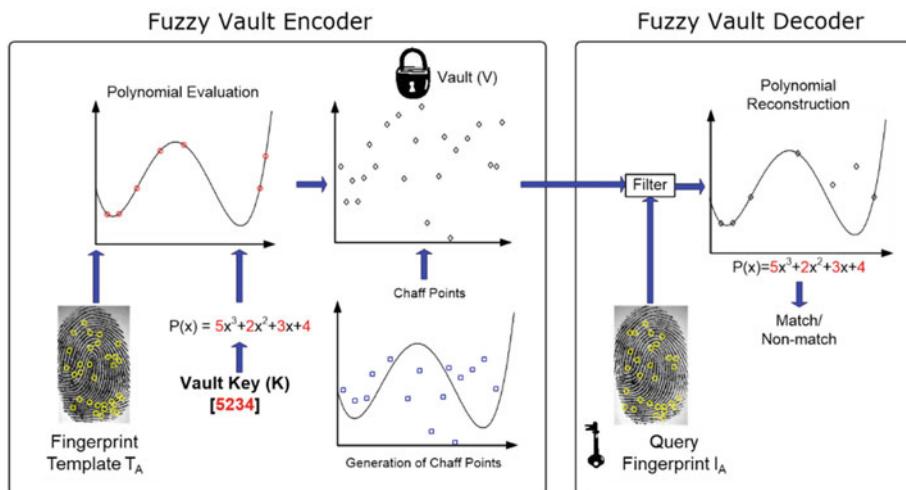


Fig. 9.25 Securing a fingerprint minutiae template using fuzzy vault

metric for fuzzy vault, binary representations with Hamming distance metric for fuzzy commitment) that is appropriate for the chosen secure sketch construct. We refer the readers to comprehensive surveys by Rathgeb and Uhl (2011) and Rane et al. (2013) for an overview of these techniques. Sutcu et al. (2007a) analyzed the secure sketch methods and identified many practical implementation issues. They caution that key generating fingerprint cryptosystems, in general, must balance between *key stability* and *key entropy*. Key stability denotes the extent to which the key generated from the fingerprint data is repeatable. Key entropy relates to the number of possible keys that can be generated. If a method generates the same key for all the fingers, then the stability is high, but the entropy is zero, leading to high false match rate. On the other extreme, if a method generates different keys for different impressions of the same finger, the scheme has high entropy but no stability, leading to high false non-match rate. In practice, it is difficult to simultaneously achieve high key entropy and high key stability from the existing key generation methods (Jain et al., 2008).

9.6.5 Feature Adaptation

A traditional fingerprint system accounts for intra-subject variations in two ways. Firstly, the feature extraction algorithm attempts to extract an invariant representation from the noisy fingerprint impressions (see Chap. 3). Secondly, the matching algorithm is designed to further suppress the effect of intra-subject variations and focus only on features that are distinctive across individuals (see Chap. 4). Template protection schemes generally

require the use of simple distance metrics such as Hamming distance or a measure of set difference to compute the similarity between biometric features (Dodis et al., 2008). Consequently, the burden of handling intra-subject variations observed in the biometric samples shifts completely to the feature extraction stage. For example, an accurate fingerprint matcher not only handles missing and spurious minutiae, but also other intra-subject variations like rotation, translation, and non-linear distortion (see Fig. 9.26). When this matcher is replaced by a simple set difference metric (that accounts for only missing and spurious minutiae) in a fingerprint cryptosystem, it becomes imperative to represent the extracted minutiae in a form that is invariant to rotation, translation, and non-linear distortion without affecting their distinctiveness. Failure to do so will naturally lead to significant degradation in the recognition performance.

Rather than developing new invariant feature extractors, which on its own is one of the fundamental problems in fingerprint recognition, researchers working on fingerprint template protection often implement a *feature adaptation* step on top of the original feature extractor. It must be emphasized that feature adaptation is not the same as feature transformation. In feature transformation, the goal is to obtain an irreversible and revocable template. In contrast, adapted templates need not satisfy the irreversibility and revocability properties. Instead, feature adaptation schemes are designed to satisfy one or more of the following three objectives: (i) minimize intra-subject variations without diluting their distinctiveness, (ii) represent the original features in a simplified form, and (iii) avoid

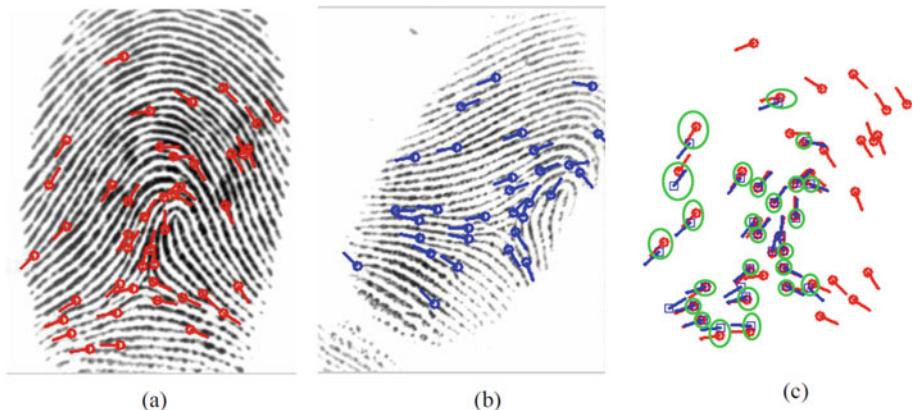


Fig. 9.26 Complexity in fingerprint minutiae matching. **a** and **b** are two fingerprint images from the same finger with minutiae features marked on them. The two minutiae sets after global alignment are shown in **c**. Apart from missing and spurious minutiae that can be captured well using the set difference metric, one can observe that the matching minutiae (marked by green ellipses) are not perfectly aligned due to non-linear distortion. This explains why a simple set difference metric is unlikely to provide accurate recognition. © IEEE. Reprinted, with permission, from Nandakumar and Jain (2015)

the need for side information (e.g., alignment parameters). While a feature transformation scheme may employ feature adaptation in the process of securing the template, the converse is not true.

The simplest and most common feature adaptation strategy is quantization and reliable component (feature) selection. A typical example is the quantization of fingerprint minutiae location and orientation features and selection of good quality minutiae (Nandakumar et al., 2007) when designing a fingerprint cryptosystem. Though the process of quantization and feature selection reduces intra-subject variations, it is also likely to decrease inter-subject variations. Thus, the challenge is to strike an optimum balance between reducing intra-subject variations and preserving inter-subject variations. Moreover, if quantization and reliable component selection is user-specific, the quantization parameters and selected components need to be stored as auxiliary data, which is likely to decrease the irreversibility and unlinkability of the protected biometric reference (Kelkboom et al., 2009).

Other strategies for feature adaptation include *embedding* and *alignment-free representation*. In embedding, the goal is to obtain a new representation for the given fingerprint features so that simple distance metrics (e.g., Hamming distance or set difference) can be used to compare fingerprint samples in the modified representation space. Conversion of a real/complex vector or point set into a fixed-length binary string is an example of biometric embedding. For instance, the Detection Rate Optimized Bit Allocation (DROBA) scheme (Chen et al., 2009) proposes an adaptive bit allocation strategy for embedding a real vector as a binary string. Techniques for converting unordered point sets (especially fingerprint minutiae) into fixed-length binary strings include local point aggregates (Nagar et al., 2010b) and spectral minutiae representation (Xu et al., 2009).

In contrast to embedding, the objective of an alignment-free representation is to generate templates that can be directly matched without the need for any alignment parameters. A possible solution to the problem of fingerprint alignment is the use of local minutiae structures, which consist of features that characterize the relative information between two or more minutiae (e.g., distance between two minutiae) (Cappelli et al., 2010). Since such features are relative, they are invariant to global rotation and translation of the fingerprint, and hence no priori alignment is needed before matching. An additional benefit is that such features are robust to nonlinear distortion. However, if the matching is based only on the local minutiae information and the global spatial relationships between minutiae are ignored, some degradation in the recognition accuracy may occur.

The simplest local minutiae structure is based on minutia pairs, where the distance between the pair and the orientation of each minutia with respect to the line connecting them can be used as the invariant attributes (Boult et al., 2007). The most commonly used local minutiae structure is the minutia triplet, where relative features (distances and angles) are computed from combinations of three minutiae. Rather than defining the local neighborhood based on a fixed number of minutiae, it is also possible to construct a local descriptor by considering all minutiae that fall within a fixed radius of a minutia point.

An example of this latter approach is the Minutia Cylinder Code (MCC) introduced in Sect. 4.4.3 (Cappelli et al., 2010). Note that it is also possible to binarize the MCC to get a fixed-length binary string describing each minutia point.

Though a significant amount of research effort has been devoted towards feature adaptation, three main issues that remained unresolved are:

1. Existing feature adaptation techniques invariably result in the loss of some discriminatory information leading to lower recognition performance. A possible reason for this phenomenon is that most of these techniques focus only on minimizing intra-subject variations while ignoring the need to preserve inter-subject variations. Hence, there is a strong need for distance-preserving feature adaptation strategies.
2. There is a de-coupling between the feature adaptation strategy and the template protection technique. For instance, the error correction scheme used in a biometric cryptosystem may have the ability to correct a limited number of errors. Since this error correction capability implicitly determines the system threshold, the feature adaptation scheme must be designed such that the number of errors between samples of the same user falls below this threshold, while the number of errors encountered during impostor comparisons is greater than the error correction capability. A feature adaptation scheme that is designed in isolation may not satisfy the above requirement. Alternatively, one can argue that it may be better to design a biometric template protection scheme that directly secures the template in its original representation rather than attempting to adapt the template to fit the template protection scheme.
3. Finally, the statistical properties of the adapted features are seldom given attention in the design of a feature adaptation scheme. For example, consider the case of a feature adaptation scheme generating a binary string as output. Apart from having low intra-subject variations and high distinctiveness, it would be ideal if the resulting binary string is uniformly random (i.e., has high entropy). Such a representation is likely to have better non-invertibility properties when it is eventually secured using a biometric cryptosystem. However, the design of such feature adaptation strategies is still an open research problem.

One of the most promising recent developments in the design of invariant feature representations for fingerprint is the DeepPrint approach introduced in Sect. 4.6.5 (Engelsma et al., 2021), which uses a combination of deep learning and fingerprint domain knowledge to extract compact fixed-length fingerprint representations. The most important advantage of this representation is that all the tricky issues in fingerprint matching such as alignment, non-linear distortion, etc., are taken care of when generating the DeepPrint. Consequently, matching reduces to a simple inner product between the template and query feature representation. It may be further possible to binarize this fixed-length representation to obtain a binary string that can be secured using known fingerprint cryptosystem frameworks.

9.6.6 Challenges and Open Issues

Most of the existing fingerprint template protection techniques do not satisfy all the desired template protection requirements in practice. As an example, consider the results published by the FVC-onGoing competition (see Sect. 4.7.2). Nine algorithms were able to achieve an equal error rate (EER) of less than 0.3% on the FVC-STD-1.0 benchmark dataset when operating without any template protection. On the other hand, the lowest EER achieved by a fingerprint verification system with template protection on the same dataset was 1.54%, which is more than 5 times higher. Reduction in accuracy was also observed during independent testing of template protection algorithms (Gafurov et al., 2013).

Even if we assume that a small degradation in the recognition performance is acceptable in some applications, it is imperative to precisely quantify (in terms of bits) the irreversibility and unlinkability of the protected biometric reference. This is necessary to benchmark the utility of a fingerprint template protection scheme. In cryptography, “security strength” (measure of the computational effort required to break a cryptosystem using the most efficient known attack) is one of the metrics used to compare different cryptosystems. It is well-known that an AES system with a 128-bit key or a RSA cryptosystem with a 3072-bit key can provide a security strength of approximately 128 bits (Barker, 2020). However, there is no consensus within the biometrics community on analogous metrics that can be used to measure the non-invertibility, revocability, and non-linkability properties of biometric template protection algorithms as well as the methods to compute these metrics. Efforts to standardize these metrics are still in progress (Rane, 2014). Consequently, practical template protection schemes neither have proven security guarantees nor do they achieve satisfactory recognition performance. This explains why despite 25 years of research, operational biometric systems usually do not go beyond encrypting the template using standard encryption techniques and/or storing them in secure hardware.

Apart from the issue of standardizing security metrics, the following unresolved challenges also need to be addressed to bridge the gap between the theory and practice of fingerprint template protection.

- The foremost issue in biometric template protection is the design of feature extractors, which not only need to extract highly robust and distinctive features, but also represent them in a simplified form (e.g., a fixed-length binary string) that is suitable for applying the template protection construct.
- The trade-off between matching accuracy and irreversibility can be solved by understanding the statistical distribution of fingerprint features and designing template protection schemes that are appropriate for the underlying feature distribution. For example, it is well-known that minutiae locations in a fingerprint (Su & Srihari, 2010) are neither independent nor do they follow a uniformly random distribution. This inherent redundancy in the minutiae features could be exploited to handle intra-subject

variations without compromising on inter-subject variations. In many fingerprint cryptosystems, the template is protected by adding noise to the true fingerprint data. In this case, knowledge of the feature distribution could be useful in selecting the appropriate noise distribution. Modeling the fingerprint feature distribution is also required for obtaining realistic estimates for the irreversibility and unlinkability of a protected biometric reference. If the biometric feature distribution is known, it may be possible to formulate biometric template protection as an optimization problem and systematically find solutions that maximize both matching accuracy and irreversibility. Thus, knowledge of the statistical distribution of fingerprint features is critical for effective fingerprint template protection. However, estimating the feature distributions remains a challenging task.

- Another way to overcome the inherent trade-off between irreversibility and matching accuracy is to develop techniques for multibiometric template protection. Since multibiometric systems accumulate evidence from more than one biometric identifier (multiple traits like fingerprint and iris or multiple fingers/irises) in order to recognize a person, they lead to a significant improvement in recognition performance. When multiple templates are secured together as a single construct, the inherent entropy of the template is also likely to be higher, thereby leading to stronger irreversibility. While a few solutions have been proposed recently for multibiometric cryptosystems (Fu et al., 2009), the fundamental challenge lies in overcoming the compatibility issues between different biometric templates and generating a combined multibiometric template from different modalities, which preserves the distinctiveness of individual templates. Advancements in feature adaptation can also play a key role in overcoming the above challenge.
- Compared to the issue of non-invertibility, the problem of ensuring unlinkability and revocability of protected fingerprint reference has not been adequately addressed in the literature. While many template protection constructs claim to provide unlinkability and revocability, a deeper analysis indicates that this is often achievable only with the involvement of an additional authentication factor (supplementary data) such as a password or secret key (Blanton & Aliasgari, 2013). It has been demonstrated that many well-known biometric cryptosystems do not generate revocable or unlinkable templates (Wang et al., 2012; Blanton & Aliasgari, 2013; Boyen, 2004 and Kelkboom et al., 2011). Though feature transformation schemes are widely proclaimed as “cancelable biometrics” in acknowledgment of their strengths in achieving revocability and unlinkability, the real capability of such schemes to guarantee these two properties is still questionable if we assume that the attacker has full knowledge of the protected biometric reference and any supplementary data involved. Revocability and unlinkability of feature transformation schemes appear to depend on the difficulty in obtaining a pre-image of the transformed template. When the pre-image is easy to compute given the transformation parameters and the transformed template, it may be possible to correlate the pre-images obtained from multiple transformed templates to invert

and/or link them (Nagar et al., 2010a). Therefore, there is a critical need to develop one-way transformation functions that do not allow easy computation of a pre-image. One possible way to achieve revocability and unlinkability is to use hybrid biometric cryptosystems (Boult et al., 2007; Feng et al., 2010). Another practical solution is the use of two- or three-factor authentication protocols. However, if we assume that all the other factors except the biometric trait are available to the attacker, the advantages of such multi-factor authentication protocols vanish, and their properties are no better than those of the underlying template protection scheme.

9.7 Building a Closed Fingerprint System

To understand the necessity and design of building a closed system, one must consider the different scenarios where modules of a fingerprint system (see Sect. 1.3) reside at different locations. We discuss two of the most common scenarios:

1. All the modules are located on a single computer (i.e., on the personal computer or smartphone of an end-user, sometimes known as a client). The end-user may protect the fingerprint system by carefully monitoring its security with anti-virus and anti-malware software, so that a remote hacker cannot take control of the local operating system. However, the user still needs to consider the case when the computer becomes physically accessible to the adversary (e.g., a stolen smartphone). In a variation of this scenario, all the modules are still located on a single computer even though the computer is shared by many end-users such as employees at a pharmacy (such computers are sometimes known as interactive kiosks). Here too, an adversary (say, a malicious pharmacy employee) can have physical access to the computer (before or after it has been used by an honest employee).
2. In a client–server application, some modules are located on the client side and some modules are located on the server side. The server is usually operated and managed by an entity that does not trust the client side (personal computer of the end-user) as the end-user may be malicious or may collude with an adversary (or be coerced by an adversary).

In both the above scenarios, the security can be enforced by moving as many modules as possible on secure (i.e., tamper-resistant) hardware that cannot be accessed by an adversary even if he has physical or remote access to the computer. There are two popular approaches that have led to commercially viable solutions:

- Move only the storage module (that contains the enrollment template) and the matcher module onto a smart card that can be in the possession of the end-user (see Fig. 9.27).

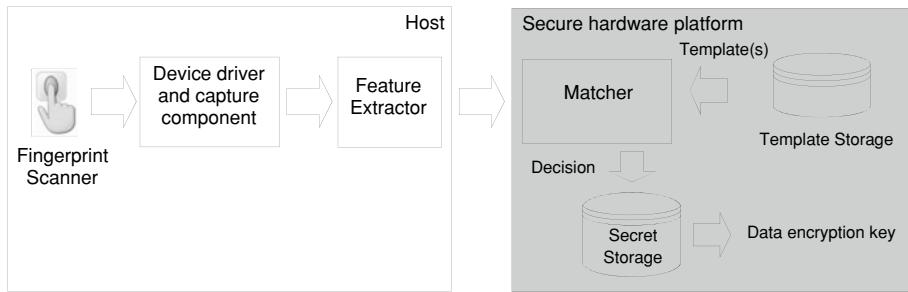


Fig. 9.27 Match-on-Card architecture; the enrolled template(s) never leave the confines of the secure hardware platform

This technique is known as *Match-on-Card* (MoC).

- Move all the modules (including feature extraction as well as the acquisition sensor) to secure hardware platforms (e.g., a hardware circuit board, smart card, or a computing chip). This approach is generally referred to as *System-on-Device* (SoD), though it is also sometimes called as *System-on-Card* or *System-on-a-Chip* (SoC), depending on the hardware platform used (see Fig. 9.28).

In a secure hardware platform, the “critical” processing occurs within a secure environment that is isolated from the operating system of the client system (i.e., the host operating system running on a laptop or smartphone). Besides the security advantage (resilience to

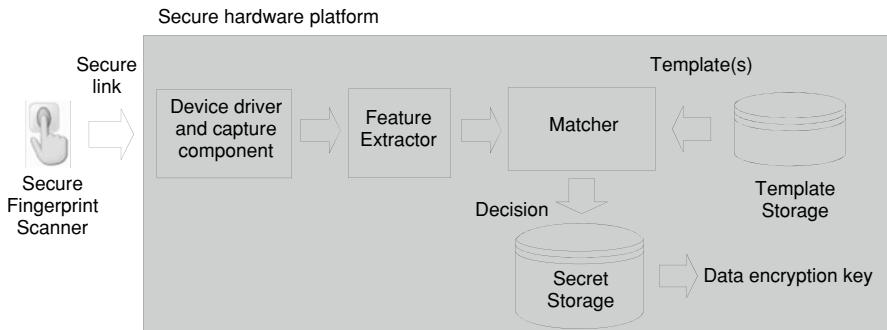


Fig. 9.28 In System-on-a-chip architecture, all the processing is done within the confines of a secure chip. Even if the scanner is physically separated from the chip, the communication channel can be protected by embedding the cryptographic keys into both the hardware (the scanner and the chip)

denial-of-service and intrusion attacks), MoC and SoD solutions also have privacy advantages. In fact, the platform is usually released to the users having full control of their own fingerprint data and there is no central enrollment database.

A secure hardware platform includes a processor (typically an embedded-class processor such as an ARMcore), workspace memory (e.g., RAM), code space memory (e.g., ROM/ EEPROM/FLASH), persistent storage (e.g., FLASH) and runs a “light” operating system. It is worth noting there is usually several orders of magnitude difference between the processing powers of a smart card and a modern PC. However, the processing gap between the system on device solutions and PCs has been narrowing significantly. For example, the secure chips used in Apple iPhones for secure biometric matching are quite powerful and their processing capabilities are at most an order of magnitude less than that of PCs. Consequently, the complexity of algorithms running on secure hardware platforms needs to be reduced, especially in the case of smart cards, which may result in some accuracy reduction. When Match-on-Card (MoC) fingerprint matching algorithms were tested as part of the MINEX II evaluation organized by NIST (Grother et al., 2007), the results showed that match-on-card algorithms were not as accurate as match-off-card (i.e., match on PC) algorithms. In particular, the best MoC algorithm exhibited a 20–40% increase in FNMR with respect to the same vendor’s implementation running on PC (at the same FMR). This is due to the simplification necessary to run them on resource limited hardware (the smart card used in MINEX II mounted a processor/microcontroller running at 8 MHz as opposed to PCs running at 2–3 GHz).

The advantage of MoC approach is that the matcher module and the template storage are fully secure. The templates can neither be modified by malicious attackers nor can they be snooped by an adversary. Once the template is transferred to a smartcard, there is no need to ever release the templates to the host. Only the matching result of the fingerprint comparison process needs to be released to the host. Finally, cryptographic keys are also stored on the smart card and hence the key management is much simplified and secured, further enhancing the system security significantly. Match-on-Card solutions are sometimes claimed to be more secure than System-on-Device solutions because of the stronger isolation of the template. However, it should be noted that even if the enrolled template cannot be obtained by an attacker, a “similar enough” template can be obtained by eavesdropping at the feature extraction module running on the non-secure host of the MoC system. The risk of such an attack should not be underestimated because eavesdropping on the host is not difficult.

In the MoC systems described above, even though the template is protected, the fingerprint feature extraction is performed on a host system, whose security may be weak and untrustworthy. This may result in intrusion and denial-of-service attacks. These vulnerabilities can be addressed by moving the remaining modules, that is, the feature extractor and possibly even the fingerprint sensor to a secure hardware platform. When the target hardware platform includes the fingerprint sensor, the solution is referred to as System-on-Device (SoD): this is the approach being proposed by some card manufacturers for

large-scale payment applications (see Sect. 2.8). When the target is a special smart card or a secure chip without the sensor, the architecture is called System-on-Card or System-on-a-Chip (SoC). As a result, no fingerprint data travels outside the secure space (see Fig. 9.28). Hence, the only remaining potential threat is a presentation attack at the user interface/sensor. The resulting system is slightly more expensive than MoC systems because the feature extractor requires a more powerful processor as well as more memory than the fingerprint matcher.

9.8 Summary

With the increasing deployment of fingerprint systems in various commercial and government applications, security of fingerprint systems itself is of growing concern to system developers, organizations deploying these systems, and the general public. Fingerprint vendors are rapidly adopting various technologies to address some of these vulnerabilities. Techniques for presentation attack detection and template protection are active areas of research.

The vulnerability of fingerprint recognition systems to presentation attacks has become a major concern, especially in the light of many well-publicized attacks reported in the media. To minimize the vulnerability of fingerprint recognition systems to such attacks, both hardware-based and software-based approaches have been proposed by the research community. While hardware-based solutions utilize the characteristics of vitality for the PA detection, the need for additional hardware is a major impediment. On the other hand, software-based solutions use a static fingerprint image or temporal variations observed in successive frames to differentiate presentation attacks from bonafide fingerprints. Software-based solutions are promising and attractive because they eschew the need for additional hardware and facilitate easy upgradation of the spoof detection capability. The advent of deep neural network-based software solutions has significantly furthered the state-of-the-art accuracy in presentation attack detection for known materials. The challenge now lies in generalizing the state-of-the-art approaches to detect attacks based on unseen presentation attack instruments and fingerprint sensors.

While fingerprint template protection has been an active research topic over the last 25 years, existing solutions are still far from gaining practical acceptance. One reason could be the success of closed fingerprint systems, especially in the era of smartphones. Another key reason for this failure is the unacceptable degradation in the recognition performance combined with unprovable security claims. Design of invariant fingerprint representations with high entropy is one of the key enablers for bridging this gap. Furthermore, standardized metrics are required for measuring the security properties of a template protection scheme, especially irreversibility. Systematic formulation of such metrics and methodologies to compute them, followed by independent benchmarking of template protection algorithms based on these metrics will greatly enhance the public confidence in

fingerprint template protection technologies. Finally, practical solutions must be devised to ensure revocability and unlinkability of protected fingerprint templates.

There is a greater need for template security in scenarios where the fingerprint data is stored in centralized repositories. Such databases are commonplace in large-scale identification systems (e.g., India's Aadhaar program, Office of Biometric Identity Management (formerly US-VISIT) program). However, almost all existing template protection techniques have been designed for the authentication use-case (one-to-one verification) as opposed to identification (one-to-many matching). While it is a pragmatic approach to start with one-to-one verification, it is not clear if such techniques can be scaled up to meet the requirements of an identification system. In identification systems, the false positive identification rate increases linearly with the number of enrolled subjects. The accuracy and throughput of existing template protection techniques cannot meet the requirements of identification systems with a large population. The only notable exception is the fully homomorphic encryption approach, which is appropriate for the identification scenario provided the decision module is isolated.

Template protection techniques address crucial security issues such as revocability of compromised templates and preventing the use of the same fingerprint data across different authentication systems. However, if an adversary is successful in intruding into a fingerprint system, it is essential to have a recovery mechanism that prevents the adversary from intruding again. It is also critical that a single breach into one system does not make it easier for an adversary to breach another system. A related issue is how to revoke and re-issue a protected biometric reference without re-enrolling the user, which is often impractical. These issues can be addressed by creating an entity similar to public key infrastructure, which can create, manage, and revoke biometric information.

It is important to recognize that foolproof fingerprint recognition systems simply may not exist. Security is a risk management strategy to identify, control, eliminate, or minimize uncertain events that may adversely affect system resources and information assets. The security requirements of a fingerprint system will depend on the threat model of the application where it would be deployed and the related cost–benefit analysis.

References

- Aadhaar Program. (2021). Unique identification authority of India: Dashboard. Government of India. Retrieved July, 2021 from https://uidai.gov.in/aadhaar_dashboard/.
- Acar, A., Aksu, H., Selcuk Uluagac, A., & Conti, M. (2018) A survey on homomorphic encryption schemes: Theory and implementation. *ACM Computing Surveys*, 51(4), 1–35.
- Agassy, M., Castro, B., Lerner, A., Rotem, G., Galili, L., & Altman, N. (2019). Liveness and spoof detection for ultrasonic fingerprint sensors. US Patent 10262188.
- Anderson, R. J. (1994). Why cryptosystems fail. *Communications of the ACM*, 37(11), 32–40.
- Antonelli, A., Cappelli, R., Maio, D., & Maltoni, D. (2006a). Fake finger detection by skin distortion analysis. *IEEE Transactions on Information Forensics and Security*, 1(3), 360–373.

- Antonelli, A., Cappelli, R., Maio, D., & Maltoni, D. (2006b). A new approach to fake finger detection based on skin distortion. In *Proceedings of International Conferences on Biometrics* (pp. 221–228).
- ANSI/NIST-ITL 1–2011. (2015). NIST, Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information, update 2015 of NIST Special Publication 500–290e3.
- Apple Inc. (2021). Apple Platform Security. Retrieved July, 2021, from <https://support.apple.com/en-sg/guide/security/welcome/web>.
- Arora, S. S., Cao, K., Jain, A. K., & Paultre, N. G. (2016). Design and fabrication of 3D fingerprint targets. *IEEE Transactions on Information Forensics and Security*, 11(10), 2284–2297.
- Baldisserra, D., Franco, A., Maio, D., & Maltoni, D. (2006). Fake fingerprint detection by odor analysis. In *Proceedings of International Conferences on Biometrics* (pp. 265–272).
- Barker, E. (2020). Recommendation for key management. NIST Special Publication 800-57.
- BBC News. (2013). Doctor ‘used silicone fingers’ to sign in for colleagues. Retrieved July 2021, from <https://www.bbc.com/news/world-latin-america-21756709>.
- Blanton, M., & Aliasgari, M. (2013). Analysis of reusability of secure sketches and fuzzy extractors. *IEEE Transactions on Information Forensics and Security*, 8(9), 1433–1445.
- Bringer, J., Chabanne, H., & Patey, A. (2013). Privacy-preserving biometric identification using secure multiparty computation: An overview and recent trends. *IEEE Signal Processing Magazine*, 30(2), 42–52.
- Boult, T. E., Scheirer, W. J., & Woodworth, R. (2007). Revocable fingerprint biotokens: Accuracy and security analysis. In *Proceedings of International Conference on Computer Vision and Pattern Recognition*.
- Boyen, X. (2014). Reusable cryptographic fuzzy extractors. In *Proceedings of Conference on Computer and Communications Security* (pp. 82–89).
- Cao, K., & Jain, A. K. (2015). Learning fingerprint reconstruction: From minutiae to image. *IEEE Transactions on Information Forensics and Security*, 10(1), 104–117.
- Cao, K., & Jain, A. K. (2016). Hacking mobile phones using 2D printed fingerprints. *MSU Technical Report*, MSU-CSE-16-2.
- Cappelli, R., Maio, D., Lumini, A., & Maltoni, D. (2007). Fingerprint image reconstruction from standard templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9), 1489–1503.
- Cappelli, R., Ferrara, M., & Maltoni, D. (2010). Minutia cylinder-code: A new representation and matching technique for fingerprint recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12), 2128–2141.
- Chaos Computer Club. (2013). Chaos Computer Club breaks Apple TouchID. Retrieved July 2021, from <https://www.ccc.de/en/updates/2013/ccc-breaks-apple-touchid>.
- Chen, C., Veldhuis, R. N. J., Kevenaar, T. A. M., & Akkermans, A. H. M. (2009). Biometric quantization through detection rate optimized bit allocation. *EURASIP Journal on Advances in Signal Processing*, 784834.
- Chugh, T. (2020). An accurate, efficient, and robust fingerprint presentation attack detector. Ph.D. Thesis, Department of Computer Science & Engineering, Michigan State University.
- Chugh, T., Cao, K., & Jain, A. K. (2017). Fingerprint spoof detection using minutiae-based local patches. In *Proceedings of International Joint Conferences on Biometrics* (pp. 581–589).
- Chugh, T., Cao, K., & Jain, A. K. (2018). Fingerprint spoof buster: Use of minutiae-centered patches. *IEEE Transactions on Information Forensics and Security*, 13(9), 2190–2202.
- Chugh, T., & Jain, A. K. (2019). Fingerprint presentation attack detection: Generalization and efficiency. In *Proceedings of International Conferences on Biometrics* (pp. 1–8).
- Chugh, T., & Jain, A. K. (2020). Fingerprint spoof detection: Temporal analysis of image sequence. In *Proceedings of International Joint Conferences on Biometrics* (pp. 1–10).

- Chugh, T., & Jain, A. K. (2021). Fingerprint spoof detector generalization. *IEEE Transactions on Information Forensics and Security*, 16(1), 42–55.
- CJIS. (2015). FBI's criminal justice information services division, latent and forensic support unit. In *Altered fingerprints: A challenge to law enforcement identification efforts Spotlights*. Retrieved July 2021, from <https://leb.fbi.gov/spotlights/forensic-spotlight-altered-fingerprints-a-challenge-to-law-enforcement-identification-efforts>.
- Cukic, B., & Bartlow, N. (2005). Biometric system threats and countermeasures: A risk based approach. In *Proceedings of Biometric Consortium Conference*.
- Cummins, H. (1935). Attempts to alter and obliterate finger-prints. *Journal of Criminal Law and Criminology*, 25(6), 982–991.
- Darlow, L. N., Webb, L., & Botha, N. (2016). Automated spoof-detection for fingerprints using optical coherence tomography. *Applied Optics*, 55(13), 3387–3396.
- Dodis, Y., Ostrovsky, R., Reyzin, L., & Smith, A. (2008). Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM Journal on Computing*, 38(1), 97–139.
- Echizen, I., & Ogane, T. (2018). Biometric jammer: Method to prevent acquisition of biometric information by surreptitious photography on fingerprints. *IEICE Transactions on Information and Systems*, E101-D(1), 2–12.
- Ellingsgaard, J., & Busch, C. (2017). Altered fingerprint detection. In M. Tistarelli & C. Champod (Eds), *Handbook of biometrics for forensic science*. Springer, Cham.
- Engelsma, J. J., Arora, S. S., Jain, A. K., & Paulte, N. G. (2018). Universal 3D wearable fingerprint targets: Advancing fingerprint reader evaluations. *IEEE Transactions on Information Forensics and Security*, 13(6), 1564–1578.
- Engelsma, J. J., Cao, K., & Jain, A. K. (2019). RaspiReader: Open source fingerprint reader. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(10), 2511–2524.
- Engelsma, J. J., & Jain, A. K. (2019). Generalizing fingerprint spoof detector: Learning a one-class classifier. In *Proceedings of IEEE International Conferences on Biometrics* (pp. 1–8).
- Engelsma, J. J., Cao, K., & Jain, A. K. (2021). Learning a fixed-length fingerprint representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(6), 1981–1997.
- Feng, J., & Jain, A. K. (2011). Fingerprint reconstruction: From minutiae to phase. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(2), 209–223.
- Feng, Y. C., Yuen, P. C., & Jain, A. K. (2010). A hybrid approach for generating secure and discriminating face template. *IEEE Transactions on Information Forensics and Security*, 5(1), 103–117.
- Ferrara, M., Maltoni, D., & Cappelli, R. (2012). Noninvertible minutia cylinder-code representation. *IEEE Transactions on Information Forensics and Security*, 7(6), 1727–1737 (2012).
- Ferrara, M., Cappelli, R., & Maltoni, D. (2017). On the feasibility of creating double-identity fingerprints. *IEEE Transactions on Information Forensics and Security*, 12(4), 892–900.
- Franco, A., & Maltoni, D. (2007). Fingerprint synthesis and spoof detection. In N. K. Ratha, & V. Govindaraju (Eds), *Advances in biometrics: Sensors, algorithms and systems*. Springer.
- Fu, B., Yang, S., Li, J., & Hu, D. (2009). Multibiometric cryptosystem: Model structure and performance analysis. *IEEE Transactions on Information Forensics and Security*, 4(4), 867–882.
- Gafurov, D., Yang, B., Bours, P., & Busch, C. (2013). Independent performance evaluation of pseudonymous identifier fingerprint verification algorithms. In *Proceedings of Interenational Conferences on Image Analysis and Recognition*.
- Gentry, C. (2009). Fully homomorphic encryption using ideal lattices. In *Proceedings of Symposium on Theory of Computing* (pp. 169–178).
- González-Soler, L. J., Gomez-Barrero, M., Chang, L., Pérez-Suárez, A., & Busch, C. (2021). Fingerprint presentation attack detection based on local features encoding for unknown attacks. *IEEE Access*, 9, 5806–5820.

- Grosz, S. A., Chugh, T., & Jain, A. K. (2020). Fingerprint presentation attack detection: A sensor and material agnostic approach. In *Proceedings of International Joint Conferences on Biometrics* (pp. 1–10).
- Grother, P., Salamon, W., Watson, C., Indovina, M., & Flanagan, P. (2007). MINEX II: Performance of fingerprint match-on-card algorithms. NIST Interagency Report 7477.
- Hern, A. (2014). Hacker fakes German minister's fingerprints using photos of her hands. *The Guardian*. Retrieved July 2021, from <https://www.theguardian.com/technology/2014/dec/30/hacker-fakes-german-ministers-fingerprints-using-photos-of-her-hands>.
- Heussner, K. M. (2009). Surgically altered fingerprints help woman evade immigration. *ABC News*. Retrieved July, 2021, from <https://abcnews.go.com/Technology/GadgetGuide/surgically-altered-fingerprints-woman-evide-immigration/story?id=9302505>.
- Hill, C. J. (2001). Risk of masquerade arising from the storage of biometrics. Bachelor of Science Thesis, The Department of Computer Science, Australian National University.
- IARPA ODIN Program. (2016). Office of the Director of National Intelligence, IARPA, “Odin,” IARPA-BAA-16-04 (Thor). Retrieved July, 2021, from <https://www.iarpa.gov/index.php/research-programs/odin/odin-baa>.
- Ignatenko, T., & Willems, F. M. J. (2009). Biometric systems: Privacy and secrecy aspects. *IEEE Transactions on Information Forensics and Security*, 4(4), 956–973.
- Ignatenko, T., & Willems, F. M. J. (2010). Information leakage in fuzzy commitment schemes. *IEEE Transactions on Information Forensics and Security*, 5(2), 337–348.
- ISO/IEC 19794-3. (2006). ISO, “ISO/IEC 19794-3:2006 – Information technology – Biometric data interchange formats – Part 3: Finger pattern spectral data”. Retrieved July, 2021, from <https://www.iso.org/standard/38747.html>.
- ISO/IEC 19794-2. (2011). ISO, “ISO/IEC 19794-2:2011 – Information technology – Biometric data interchange formats – Part 2: Finger minutiae data”. Retrieved July, 2021, from <https://www.iso.org/standard/50864.html>.
- ISO/IEC 19794-4 (2011). ISO, “ISO/IEC 19794-4:2011 – Information technology – Biometric data interchange formats – Part 4: Finger image data”. Retrieved July, 2021, from <https://www.iso.org/standard/50866.html>.
- ISO/IEC 24745 (2011). ISO, “ISO/IEC 24745:2011 – Information technology – Security techniques – Biometric information protection”. Retrieved July, 2021, from <https://www.iso.org/standard/52946.html>.
- ISO/IEC 30107-1 (2016). ISO, “ISO/IEC 30107-1:2016 – Information Technology – Biometric Presentation Attack Detection – Part 1: Framework”. Retrieved July, 2021, from <https://www.iso.org/standard/53227.html>.
- ISO/IEC 19989-1/2/3 (2020). ISO, “ISO/IEC 19989:2020 – Information security – Criteria and methodology for security evaluation of biometric systems”. Retrieved July, 2021, from <https://www.iso.org/standard/72402.html>.
- Jain, A. K., Nandakumar, K., & Nagar, A. (2008). Biometric template security. *EURASIP Journal on Advances in Signal Processing, Special Issue on Advanced Signal Processing and Pattern Recognition Methods for Biometrics*, (113), 1–17.
- Juels, A., & Sudan, M. (2002). A fuzzy vault scheme. In *Proceedings of International Symposium on Information Theory*.
- Juels, A., & Wattenberg, M. (1999). A fuzzy commitment scheme. In *Proceedings of Conference on Computer and Communications Security* (pp. 28–36).
- Kelkboom, E. J. C., de Groot, K. T. J., Chen, C., Breebaart, J., & Veldhuis, R. N. J. (2009). Pitfall of the detection rate optimized bit allocation within template protection and a remedy. In *Proceedings of International Conferences on Biometrics: Theory, Applications, and Systems* (pp. 1–8).

- Kelkboom, E. J. C., Breebaart, J., Kevenaar, T. A. M., Buhan, I., & Veldhuis, R. N. J. (2011). Preventing the decodability attack based cross-matching in a fuzzy commitment scheme. *IEEE Transactions on Information Forensics and Security*, 6(1), 107–121.
- Kong, A., Cheung, K., Zhang, D., Kamel, M., & You, J. (2006). An analysis of BioHashing and its variants. *Pattern Recognition*, 39(7), 1359–1368.
- Korkzani, S. (2016). How MSU researchers unlocked a fingerprint-secure smartphone to help police with homicide case. *The State News*. Retrieved July, 2021, from <http://statenews.com/article/2016/08/how-msu-researchers-unlocked-a-fingerprint-secure-smartphone-to-help-police-with-homicide-case>.
- Lai, L., Ho, S. W., & Poor, H. V. (2011). Privacy-security trade-offs in biometric security systems. *IEEE Transactions on Information Forensics and Security*, 6(1), 122–151.
- Li, S., & Kot, A. C. (2012). An improved scheme for full fingerprint reconstruction. *IEEE Transactions on Information Forensics and Security*, 7(6), 1906–1912.
- Marasco, E., & Ross, A. (2015). A survey on antispoofing schemes for fingerprint recognition systems. *ACM Computing Surveys*, 47(2), 1–36.
- Marcel, S., Nixon, M. S., Fierrez, J., & Evans, N. (2019). *Handbook of biometric anti-spoofing: Presentation attack detection* (2nd ed.). Springer.
- Matsumoto, T., Matsumoto, H., Yamada, K., & Hoshino, S. (2002). Impact of artificial ‘gummy’ fingers on fingerprint systems. In *Proceedings of SPIE* (Vol. 4677, pp. 275–289).
- McGraw, G. (2006). *Software security*. Digital.
- Mopuri, K. R., Garg, U., & Venkatesh Babu, R. (2018). CNN fixations: An unraveling approach to visualize the discriminative image regions. *IEEE Transactions on Image Processing*, 28(5), 2116–2125.
- Moujahdi, C., Bebis, G., Ghouzali, S., & Rziza, M. (2014). Fingerprint shell: Secure representation of fingerprint template. *Pattern Recognition Letters*, 45, 189–196.
- Nagar, A., Nandakumar, K., & Jain, A. K. (2010a). Biometric template transformation: A security analysis. In *Proceedings of SPIE Conferences on Electronic Imaging – Media Forensics and Security II* (Vol. 7541).
- Nagar, A., Rane, S., & Vetro, A. (2010b). Privacy and security of features extracted from minutiae aggregates. In *Proceedings International Conferences on Acoustics, Speech and Signal Processing* (pp. 1826–1829).
- Nandakumar, K., & Jain, A. K. (2015). Biometric template protection: Bridging the performance gap between theory and practice. *IEEE Signal Processing Magazine*, 32(5), 88–100.
- Nandakumar, K., Jain, A. K., & Pankanti, S. (2007). Fingerprint-based fuzzy vault: Implementation and performance. *IEEE Transactions on Information Forensics and Security*, 2(4), 744–757.
- Nogueira, R. F., Lotufo, R. A., & Machado, R. C. (2016). Fingerprint liveness detection using convolutional neural networks. *IEEE Transactions on Information Forensics and Security*, 11(6), 1206–1213.
- Pala, F., & Bhanu, B. (2017). Deep triplet embedding representations for liveness detection. In B. Bhanu, & A. Kumar (Eds), *Deep learning for biometrics*. Springer.
- Patel, V. M., Ratha, N. K., & Chellappa, R. (2015). Cancelable biometrics: A review. *IEEE Signal Processing Magazine*, 32(5), 54–65.
- Plesh, R., Bahmani, K., Jang, G., Yambay, D., Brownlee, K., Swyka, T., Johnson, P., Ross, A., & Schuckers, S. (2019). Fingerprint presentation attack detection utilizing time-series, color finger-print captures. In *Proceedings of International Conferences on Biometrics* (pp. 1–8).
- Priesnitz, J., Rathgeb, C., Buchmann, N., Busch, C., & Margraf, M. (2021). An overview of touchless 2D fingerprint recognition. *EURASIP Journal on Image and Video Processing* 8, 1–28.
- Ratha, N. K., Connell, J. H., & Bolle, R. M. (2001). Enhancing security and privacy in biometrics-based authentication systems. *IBM Systems Journal*, 40(3), 614–634.

- Ratha, N. K., Chikkerur, S., Connell, J. H., & Bolle, R. M. (2007). Generating cancelable fingerprint templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4), 561–572.
- Rathgeb, C., & Uhl, A. (2011). A survey on biometric cryptosystems and cancelable biometrics. *EURASIP Journal on Information Security*, 2011(1), 1–25.
- Rane, S. (2014). Standardization of biometric template protection. *IEEE MultiMedia*, 21(4), 94–99.
- Rane, S., Wang, Y., Draper, S. C., & Ishwar, P. (2013). Secure biometrics: Concepts, authentication architectures, and challenges. *IEEE Signal Processing Magazine*, 30(5), 51–64.
- Rattani, A., Scheirer, W. J., & Ross, A. (2015). Open set fingerprint spoof detection across novel fabrication materials. *IEEE Transactions on Information Forensics and Security*, 10(11), 2447–2460.
- Roberts, C. (2007). Biometric attack vectors and defences. *Computers and Security*, 26(1), 14–25.
- Ross, A., Shah, J., & Jain, A. K. (2007). From template to image: Reconstructing fingerprints from minutiae points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4), 544–560.
- Rowe, R. K., Nixon, K. A., & Butler, P. W. (2008). Multispectral fingerprint image acquisition. In N. K. Ratha & V. Govindaraju (Eds), *Advances in biometrics*. Springer, London.
- Roy, A., Memon, N., & Ross, A. (2017). MasterPrint: Exploring the vulnerability of partial fingerprint-based authentication systems. *IEEE Transactions on Information Forensics and Security*, 12(9), 2013–2025.
- Schneier, B. (1996). *Applied cryptography*. Wiley.
- Shimamura, T., Morimura, H., Shimoyama, N., Sakata, T., Shigematsu, S., Machida, K., & Nakaniishi, M. (2008). A fingerprint sensor with impedance sensing for fraud detection. In *Proceedings of International Solid-State Circuits Conference - Digest of Technical Papers*.
- Sousedík, C., & Busch, C. (2014). Presentation attack detection methods for fingerprint recognition systems: A survey. *IET Biometrics*, 3(4), 219–233.
- Soutar, C. (2002). Biometric system security. In *Secure – The silicon trust magazine* (Vol. 5).
- Soutar, C. (2004). Security considerations for the implementation of biometric systems. In N. Ratha, & R. Bolle (Eds), *Automatic fingerprint recognition systems*. Springer.
- Su, C., & Srihari, S. (2010). Evaluation of rarity of fingerprints in forensics. In *Proceedings of Advances in Neural Information Processing Systems* (pp. 1207–1215).
- Sutcu, Y., Li, Q., & Memon, N. (2007a). Protecting biometric templates with sketch: Theory and practice. *IEEE Transactions on Information Forensics and Security*, 2(3), 503–512.
- Sutcu, Y., Sencar, H. T., & Memon, N. (2007b). A geometric transformation to protect minutiae-based fingerprint templates. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification IV*.
- Tabassi, E., Chugh, T., Deb, D., & Jain, A. K. (2018). Altered fingerprints: Detection and localization. In *Proceedings of International Conferences on Biometrics Theory, Applications and Systems* (pp. 1–9).
- TABULA RASA Program. (2013). European Commission, Trusted Biometrics under Spoofing Attacks (TABULA RASA). Retrieved July, 2021, from <http://www.tabularasa-euproject.org/>.
- Tan, B., & Schuckers, S. (2006). Liveness detection for fingerprint scanners based on the statistics of wavelet signal processing. In *Proceedings of CVPR Workshop on Biometrics*.
- Teoh, A., Goh, A., & Ngo, D. (2006). Random multispace quantization as an analytic mechanism for BioHashing of biometric and random identity inputs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12), 1892–1901.
- Tolosana, R., Gomez-Barrero, M., Busch, C., & Ortega-Garcia, J. (2020). Biometric presentation attack detection: Beyond the visible spectrum. *IEEE Transactions on Information Forensics and Security*, 15, 1261–1275.

- Tulyakov, S., Farooq, F., Mansukhani, P., & Govindaraju, V. (2007). Symmetric hash functions for secure fingerprint biometric systems. *Pattern Recognition Letters*, 28(16), 2184–2189.
- Vidyut (2018). Cloned thumb prints used to spoof biometrics and allow proxies to answer online Rajasthan Police exam. *Medianama*. Retrieved July, 2021, from <https://www.medianama.com/2018/03/223-cloned-thumb-prints-used-to-spoof-biometrics-and-allow-proxies-to-answer-online-rajasthan-police-exam/>.
- Wang, Y., Rane, S., Draper, S. C., & Ishwar, P. (2012). A theoretical analysis of authentication, privacy, and reusability across secure biometric systems. *IEEE Transactions on Information Forensics and Security*, 7(6), 1825–1840.
- Weiss, B., Cranley, E., & Pasley, J. (2020). These are the fugitives on the FBI's 10 most wanted list – and how they got there. *Business Insider*. Retrieved July, 2021, from <https://www.businessinsider.com/fbi-10-most-wanted-criminals-list-2017-11>.
- Xu, H., Veldhuis, R. N. J., Bazen, A. M., Kevenaar, T. A. M., Akkermans, T. A. H. M., & Gokberk, B. (2009). Fingerprint verification using spectral minutiae representations. *IEEE Transactions on Information Forensics and Security*, 4(3), 397–409.
- Yambay, D., Ghiani, L., Marcialis, G. L., Roli, F., & Schuckers, S. (2019). Review of fingerprint presentation attack detection competitions. In S. Marcel, M. Nixon, J. Fierrez, & N. Evans (Eds), *Handbook of biometric anti-spoofing*. Springer.
- Yau, W. Y., Tran, H. L., & Teoh, E. K. (2008). Fake finger detection using an electrotactile display system. In *Proceedings of International Conferences on Control, Automation, Robotics and Vision* (pp. 962–966).
- Yoon, S., Feng, J., & Jain, A. K. (2012). Altered fingerprints: Analysis and detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(3), 451–464.
- Zhang, Y., Tian, J., Chen, X., Yang, X., & Shi, P. (2007). Fake finger detection based on thin-plate spline distortion model. In *Proceedings of International Conferences on Biometrics* (pp. 742–749).
- Zhang, Y., Shi, D., Zhan, X., Cao, D., Zhu, K., & Li, Z. (2019). Slim-ResCNN: A deep residual convolutional neural network for fingerprint liveness detection. *IEEE Access*, 7, 91476–91487.