Advances in Computer Vision and Pattern Recognition

Ajita Rattani
Reza Derakhshani
Arun Ross *Editors*

Selfie Biometrics

Advances and Challenges

Springer

Advances in Computer Vision and Pattern Recognition

More information about this series at http://www.springer.com/series/4205

Ajita Rattani · Reza Derakhshani ·
Arun Ross
Editors

Selfie Biometrics

Advances and Challenges

Springer

Editors
Ajita Rattani
Department of Electrical Engineering
and Computer Science
Wichita State University
Wichita, KS, USA

Reza Derakhshani
Department of Computer Science
and Electrical Engineering
University of Missouri-Kansas City
Kansas City, MO, USA

Arun Ross
Department of Computer Science
and Engineering
Michigan State University
East Lansing, MI, USA

ISSN 2191-6586 ISSN 2191-6594 (electronic)
Advances in Computer Vision and Pattern Recognition
ISBN 978-3-030-26971-5 ISBN 978-3-030-26972-2 (eBook)
https://doi.org/10.1007/978-3-030-26972-2

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

Preface

Biometrics is the science of recognizing individuals based on their biological or behavioral attributes such as face, fingerprints, iris, gait, voice, or typing pattern. Biometric solutions are being increasingly incorporated in mobile devices such as smartphones in order to secure mobile phone services such as banking. In this context, "selfie biometrics" has gained increased attention from the research community and industry alike. A "selfie," by definition, is a self-portrait photograph, typically taken with a smartphone that is held in the hand or supported by a selfie stick. "Selfie biometrics" is, therefore, an authentication mechanism where a user captures images of her own biometric traits (such as face, fingerprints, or iris) by using the sensors available in the device itself. Thus, no additional hardware is required for the acquisition of biometric samples for mobile user authentication.

Recently, a number of papers have been published on the topic of selfie biometrics. This book provides the first comprehensive description of the state-of-the-art in selfie biometrics using face, ocular, fingerprint, and other modalities. This book includes an introductory chapter by the editors that summarizes the state-of-the-art in this topic, followed by individual chapters describing the various modalities that are being used for this purpose, the methods that have been developed to perform authentication using these modalities, and an analysis of the robustness, privacy, and usability of each method. Liveness detection and soft-biometrics prediction from selfie images are also covered in this book.

Overall, this book aims to present a clear understanding, recent advances, and challenges in the field of selfie biometrics. This book is suitable for final-year graduate students, postgraduate students, engineers, researchers, and academicians in the field of computer science and engineering who are engaged in various disciplines of system sciences, information security, privacy, and identity businesses. The objective of this book is also to engage researchers from academia and industry on the state-of-the-art mobile biometric research and technology and to mitigate the potential problems in real-time integration of selfie biometrics for mobile user authentication.

Unique Features

1. First comprehensive book on the state-of-the-art methods for selfie face, finger, and ocular biometrics for mobile user authentication
2. Review latest developments on privacy, security, usability, liveness detection, and soft-biometric prediction from selfie images
3. Enlist the challenges involved in the real-time integration of selfie biometrics for mobile use cases.

Audience

This book is essential reading for anyone involved in biometric-based person authentication, privacy and security, mobile security, and adversarial pattern classification. Students, researchers, practitioners, engineers, and technology consultants are the main audience. Those who are new to the field will also benefit from the introductory chapter outlining the basics for the most important terms and topics associated with selfie biometrics.

Organization

This book is organized as follows: Chapter 1 provides an overview of selfie face, finger, and ocular modalities. Further, spoofing and anti-spoofing techniques, soft-biometrics prediction, cloud-based services and challenges, and future research directions for selfie biometrics are also discussed. Part I discusses methods for selfie finger, ocular, and face biometrics for mobile user authentication. Part II discusses spoofing and anti-spoofing schemes for selfie biometrics. Part III discusses soft-biometric prediction and continuous user authentication based on selfie biometrics. Part IV discusses the framework for security, privacy, and case study on usability along with a distributed protocol for selfie biometrics.

Wichita, USA Ajita Rattani
Kansas City, USA Reza Derakhshani
East Lansing, USA Arun Ross

Contents

Part I Selfie Finger, Ocular and Face Biometrics

About the Editors

Ajita Rattani is Assistant Professor in the Department of Electrical Engineering and Computer Science at Wichita State University since 2019. Prior to this, she was Adjunct Graduate Faculty at the University of Missouri-Kansas City. She did her Postdoctoral and Ph.D. studies from Michigan State University and University of Cagliari, Italy, respectively. Her fields of research are biometrics, machine learning, deep learning, image processing, and computer vision. She is co-editor of Springer book titled *Adaptive Biometric Systems: Recent Advances and Challenges*. She has received a number of best paper awards at IEEE international conferences and is Editorial Board Member of IEEE Biometrics Council.

Reza Derakhshani is Associate Professor of Computer Science and Electrical Engineering at the University of Missouri-Kansas City. He is also Chief Scientist and technology inventor at EyeVerify (now ZOLOZ), a Kansas City biometric start-up that was acquired by Alibaba's Ant Financial in 2016. He earned his Ph.D. and master's degrees in computer and electrical engineering from West Virginia University. His research interests are in biometrics, computational imaging, and biomedical signal and image processing using computational intelligence paradigms. His work has been sponsored by private industry and various state and federal agencies and has resulted in many publications and issued the US and international patents.

Arun Ross is Professor in the Department of Computer Science and Engineering at Michigan State University. Prior to joining MSU, he was Faculty Member at West Virginia University. He is co-author of the books *Introduction to Biometrics* and *Handbook of Multibiometrics*. He is a recipient of the JK Aggarwal Prize and the Young Biometrics Investigator Award from the International Association of Pattern Recognition for his contributions to the field of pattern recognition and biometrics. He was designated Kavli Fellow by the US National Academy of Sciences by virtue of his presentation of the 2006 Kavli Frontiers of Sciences Symposium.

Chapter 1
Introduction to Selfie Biometrics

Ajita Rattani, Reza Derakhshani and Arun Ross

Abstract Traditional password-based solutions are being predominantly replaced by biometric technology for mobile user authentication. Since the inception of smartphones, smartphone cameras have made substantial progress in image resolution, aperture size, and sensor size. These advances facilitate the use of selfie biometrics such as the self-acquired face, fingerphoto, and ocular region for mobile user authentication. This chapter introduces the topic of selfie biometrics to the readers. Overview of the methods for different selfie biometrics modalities is provided. Liveness detection, soft-biometrics prediction, and cloud-based infrastructure for selfie biometrics are also discussed. Open issues and research directions are included to provide the path forward. The overall aim is to improve the understanding and advance the state-of-the-art in this field.

1.1 Mobile Biometrics

Biometrics is the science of recognizing an individual based on the inherent physical (fingerprints, iris, face, hand geometry, and palmprint) or behavioral traits (gait, voice, and signature) associated with the person [1]. A conventional biometric system operates by capturing the biometric trait of a person and comparing the acquired sample with the biometric template(s) in a database to determine the identity or to validate a claimed identity.

With the unprecedented mobile technology revolution, mobile devices have transcended from their primary communication role to all-in-one platforms for

A. Rattani (✉)
Wichita State University, Wichita, KS, USA
e-mail: ajita.rattani@wichita.edu

R. Derakhshani
University of Missouri—Kansas City, Kansas City, MO, USA
e-mail: derakhshanir@umkc.edu

A. Ross
Michigan State University, East Lansing, MI, USA
e-mail: rossarun@cse.msu.edu

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_1

shopping, entertainment, productivity, and social networking. An increasing number of individuals are accessing the internet and online services, such as e-commerce and banking, using their smartphones instead of traditional desktop computers. Although individuals are using their smartphones for sensitive applications and transactions, these devices can be easily misplaced, lost, or stolen more often than other computing devices, thereby demanding the use of effective user authentication mechanisms. Traditional methods for mobile security include the use of passwords, PINs, and screen lock patterns to restrict access to authorized users. However, these methods have many security drawbacks: They can be guessed, forgotten, stolen, or eavesdropped.

Password replacement solutions are now predominantly based on biometrics. In some cases, passcodes are used in conjunction with biometrics in a multifactor configuration. The use of biometric technology in mobile devices has been referred to as *mobile biometrics*, encompassing the sensors that acquire biometric samples as well as the associated algorithms for preprocessing, and matching the biometric samples to verify the claimed identity [2–4].

Since the inception of smartphones, smartphone *cameras* have made substantial progress. Image resolution, aperture size, and a sensor size of smartphone cameras have all improved tremendously over time. Since 2008, the megapixel count of these images has gone up from 2 to 20+; apertures have become brighter, with f/1.4 camera modules being considered; and sensor diagonal has increased from 0.25 inches to approximately 0.45 inches.[1] These advances in smartphone cameras facilitate the acquisition and integration of biometric modalities such as the face and ocular region for mobile user authentication [5–8]. Figure 1.1 shows an example of face-based mobile user authentication. This figure was taken from https://www.scnsoft.com/blog/3d-face-recognition-to-join-a-list-of-mobile-enabled-biometrics.

Other popular modalities such as fingerprint and iris that are used for mobile user authentication warrant the use of additional hardware for data acquisition. Further, behavioral biometrics such as gait/motion, keystroke, and touch/swipe analysis have also been used for user authentication in mobile devices [9, 10].

Mobile biometrics is, ubiquitously, installed in 100 percent of mobile devices, fueled by advances in mobile biometrics and rapid expansion of smartphones' market share. Figure 1.2 is a chart from Statista showing biometrics to be installed on 100% of wearables and tablets by 2020. In fact, the latest smartphones provide a range of biometric capabilities, with the most common OEM-provided modalities being the face, fingerprint, and at times iris recognition. Mobile device applications include online banking, password vaults, signing documents univocally, secure access to Web sites, and execution of administration procedures. E-commerce giant Alibaba is using facial recognition service in their mega-app Alipay Wallet.[2] MasterCard[3] has introduced user authentication based on face biometrics, and many more followed suit. Some versions of the Android mobile operating system have also used face

[1] https://petapixel.com/2017/06/16/smartphone-cameras-improved-time/.

[2] https://www.computerworld.com/article/2897117/alibaba-uses-facial-recognition-tech-for-online-payments.html.

[3] http://www.bbc.com/news/technology-35631456.

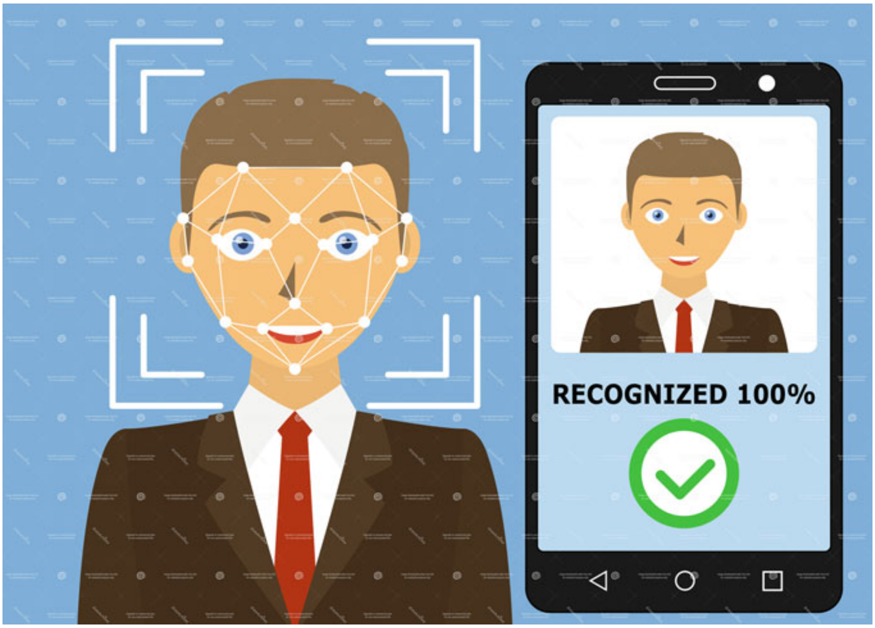


Fig. 1.1 Face biometrics for mobile user authentication

biometrics to log in users (Google has developed “Face Unlock” for Android 4.0).[4] It is reported that future versions of Android will be shipped with native support for more advanced and secure $3D$ face recognition algorithms,[5] similar to what Apple introduced under their “Face ID” moniker with iPhone X.

The applications of mobile biometrics are in border control, financial transactions, and physical and logical access control.

- **Border Control**: Passenger-friendly security is one of the primary concerns at high-volume border checkpoints such as airports. Mobile devices are exceedingly being utilized to facilitate customs and border crossings[6] to address such needs. As such, deployment of mobile devices is poised to automate the process of traveler identification and border security at checkpoints like airports and seaports in a secure yet user-friendly and private manner. Mobile passport apps are already taking advantage of modalities such as face to authenticate and process international travelers using their smartphones.[7]
- **Financial Market**: The democratization of financial services has gone hand-in-hand with the spread of mobile technologies, enabling consumers to have access

[4] https://www.technologyreview.com/s/425805/new-google-smart-phone-recognizes-your-face/.

[5] http://www.planetbiometrics.com/article-details/i/9918/desc/google-developing-3d-face-authentication/.

[6] https://www.airsidemobile.com.

[7] https://mobilepassport.us/faq.php.

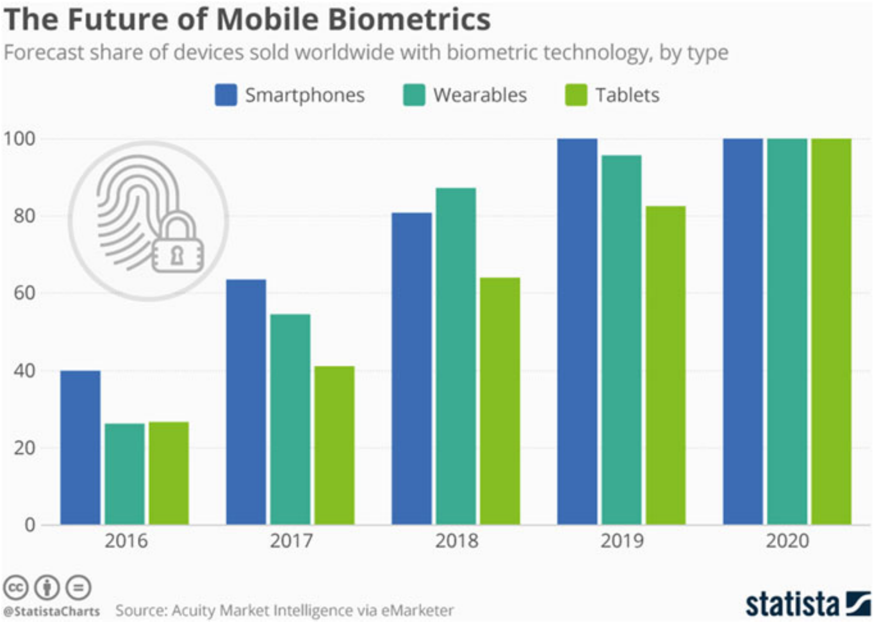


Fig. 1.2 Chart from Statista, biometrics to be installed on 100% of wearables and tablets by 2020. *Source* https://www.statista.com/chart/11122/the-future-of-mobile-biometrics/

to a wide swath of financial services without needing the traditional brick-and-mortar institutions, especially in developing markets. Examples include online shopping, micro-lending, immediate transfer of funds, or paying bills via mobile apps. Biometrics is increasingly being used to authenticate the involved parties in such transactions. Mobile wallets and other payment systems such as Apple Pay and Android Pay, along with major players like MasterCard[8] are utilizing smartphone-based biometric authentication for financial transactions.

- **Physical and Logical Access Control**: Access control is used to regulate restricted access to resources or a place. Physical and logical are the two main types of access control. While physical access control limits access to buildings, rooms, areas, and IT assets, logical access control limits connection to computer networks, system files, and data. The role of biometrics in physical and logical access control is to avoid illegal access by validating the identity of a user through biometric traits. These biometric-based access control solutions are better authentication methods compared to physical keys, key cards, and PINs because they cannot be lost, stolen, and easily compromised. Out of band authentication is one popular method where a mobile device is used to transmit the user's identity from his or her phone to a nearby logical or physical asset in need of user authentication, such as a personal computer or a smart lock.

[8] http://newsroom.mastercard.com/eu/press-releases/mastercard-makes-fingerprint-and-selfie-paymenttechnology-a-reality/.

Mobile biometrics aims to achieve conventional functionality and robustness while supporting portability, mobility, and user experience; bringing greater convenience and opportunity for deployment in a wide range of operational environments. The technology is expected to continue experiencing exponential growth due to increased consumer demand for convenient security. The Global Biometrics and Mobility Report in 2017 by Acuity Market Intelligence projected that global mobile biometric market revenues will reach 50.6 billion annually by 2022. This includes 2.7 billion biometrically enabled smart mobile devices generating 3.1 billion in biometric sensor revenue annually, 16.7 billion biometric app downloads generating 29.2 billion in annual revenues from direct purchase and software development fees, and 1.37 trillion biometrically secured payment and non-payment transactions generating 18.3 billion in annual authentication fees: http://www.acuity-mi.com/GBMR_Report.php/.

However, it is worth mentioning that classical methods for biometric recognition may not be readily adaptable to a mobile environment because of the following factors:

- Due to device mobility and operation in an uncontrolled environment, biometric samples acquired using a mobile phone's front-facing cameras are usually degraded due to factors such as specular reflection, motion blur, illumination variation, and background lighting, not to mention the inherent lower quality of front-facing cameras compared to the main back-facing modules used in smartphones. Therefore, more efficient and robust methods may be required for biometric integration in mobile devices.
- Although the computational power of mobile devices is proliferating, it still may not be sufficient for real-time operation of highly accurate and computationally costly methods for biometric authentication.
  Given that about 0.5 seconds is spent by the camera module to initialize, meter, and capture an image, an ideal biometric recognition module should take less than half a second for the whole process not to make more than a second, an essential factor in user experience.

Therefore, most of the proposed studies on mobile biometric methods have emphasized on developing computationally efficient methods (low memory and CPU impact) for accurate recognition of mobile use cases [11–13].

1.2 Selfie Biometrics

The storage and computational capability of smartphones have improved substantially over time. Figure 1.3 shows the enhancement in the storage capabilities of different models of flagship smartphones.

Chipsets from four leading vendors that power the handsets are as follows: Apple's 4-core A10 Fusion (iPhone 7/7 plus) and 6-core AI- and AR-optimized A11 Bionic

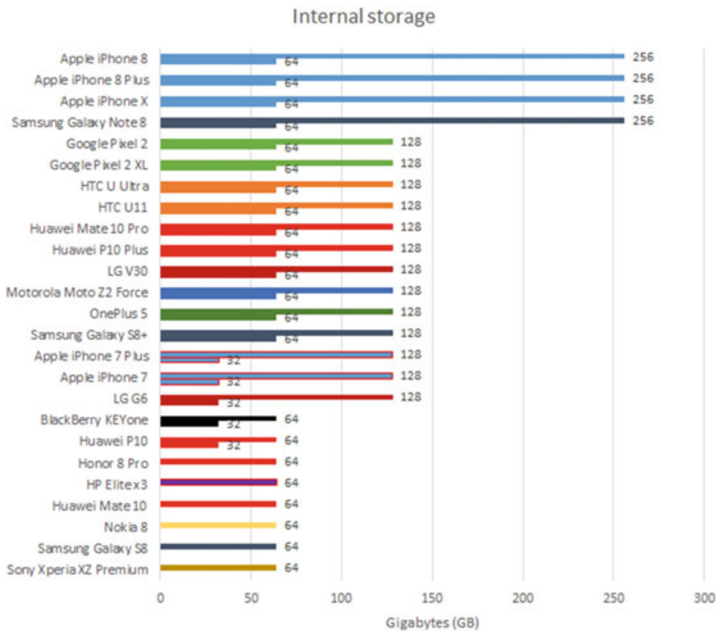


Fig. 1.3 Charts from ZDNet shows substantial improvement in storage capability of flagship smartphones. *Source* https://www.zdnet.com/article/flagship-smartphones-specs-benchmarks-and-prices-for-iphone-samsung-huawei-and-more//

(iPhone 8/8Plus/X). Samsung's 8-core Exynos 8995 in the Galaxy S8/S8+/Note 8 (worldwide versions). Qualcomm's mid-range 8-core Snapdragon 625 (BlackBerry KEYone and Motion); 4-core 820 (HP Elite x3) and 821 (HTC U Ultra, LG G6); and top-end 8-core 835 (Google Pixel 2/2XL, HTC U11+, LG V30, Moto Z2 Force, OnePlus 5T, Galaxy S8/S8+/Note 8 [US/China versions], Sony Xperia XZ Premium). HiSilicon's Kirin 960 in the Huawei P-series and Honor handsets, and the AI-optimized 8-core Kirin 970 in the new Huawei Mate 10 and 10 Pro.[9] Chart in Fig. 1.4 shows how these platforms measure up in terms of processor and graphics performance, as assessed by Primate Labs' multi-core Geekbench 4 (Gb4) and Futuremark's 3DMark Ice Storm Unlimited (ISU) benchmarks, respectively. This chart shows continuous improvement in the CPU and GPU performance over time.

The advancement in storage and computational performance, to a great extent, facilitate the use of *Selfie biometrics*. *In the context of mobile device, a selfie, by definition, is a self-portrait photograph, typically taken with a smartphone's camera while being held in hand or supported by a selfie stick. Selfie biometrics is, therefore, an authentication mechanism where a user captures images of her biometric traits (such as the face or ocular region) by using the imaging sensors available in the device itself.*

[9] https://www.zdnet.com/article/flagship-smartphones-specs-benchmarks-and-prices-for-iphone-samsung-huawei-and-more/.

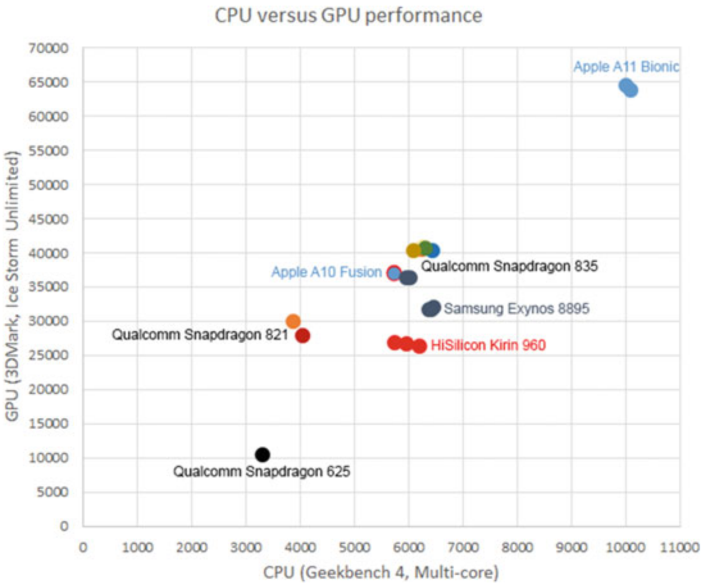


Fig. 1.4 Charts from ZDNet shows substantial improvement in computational capability of flagship smartphones. *Source* https://www.zdnet.com/article/flagship-smartphones-specs-benchmarks-and-prices-for-iphone-samsung-huawei-and-more//

The advantages of selfie biometrics include:

- **No Additional Hardware Needed**: As the mobile camera is used for selfie image acquisition, no additional hardware is needed for personal authentication in mobile devices.
- **High Acceptability and Usability**: Over 1 million selfies are taken each day globally (https://infogram.com/selfie-statistics-1g8djp917wqo2yw). Given the popularity of selfies, it is widely accepted as a means of mobile user authentication.

Challenges of selfie biometrics include intra-class variations such as poses, occlusion, low lighting, spectral reflection, and motion blur due to operation in a free mobile environment.

1.2.1 Types of Selfie Biometrics

1.2.1.1 Face

Figure 1.5 shows sample face images acquired using the front-facing camera of an iPhone 5*s*. The complete face recognition pipeline consists of selfie face acquisition, face detection, possibly normalization, and finally matching with one or more

Fig. 1.5 Example face images acquired using the front-facing camera of iPhone 5s

stored templates. Face normalization reduces the effect of intra-class variations such as lighting and poses variations through preprocessing, geometric frontalization, and registration routines. Most of the proposed studies on mobile face biometrics have emphasized developing computationally efficient methods (low memory and CPU impact) for face detection (such as optimized Viola–Jones) and recognition [5, 14–21]. Mobile face recognition methods can be broadly categorized into (a) client–server based and (b) device based [21]. In the client–server approach, face acquisition, face detection, and sometimes feature extraction routines are performed on the device side. The remaining computationally intensive tasks, such as classifier training and recognition, are performed on the server. In the device-based approach, all of the operations are performed within the device and exceedingly using secure hardware pipelines. The templates themselves are usually stored on the device, especially with native OEM implementations. However, third-party apps may store templates on secure servers via their cloud services. Of late, deep learning such as CNN solutions have been successfully ported into mobile phones, and they are working with very high accuracy and speed both on the device- and server-side [22] applications. One widely deployed commercial example is Face++ (https://www.faceplusplus.com/)).

1.2.1.2 Ocular

Ocular biometrics encompasses the imaging and use of characteristic features extracted from the eyes for personal recognition. Ocular biometric modalities in visible light have mainly focused on iris, blood vessel structures over the white of the eye (mostly due to conjunctival and episcleral layers), and the periocular region around the eye. Figure 1.7 shows an example of an eye image labeled with iris, conjunctival vasculature, and periocular region. Textural descriptors (such as LBP,

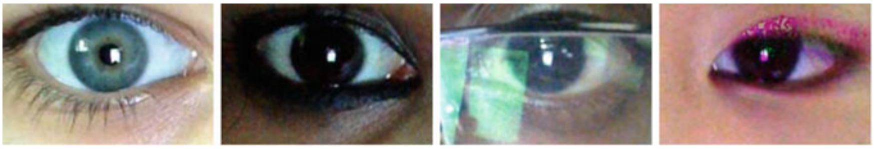


Fig. 1.6 Sample eye images acquired using iPhone 5s containing variations such as **a** light and **b** dark irides, **c** reflection, and **d** imaging artifact

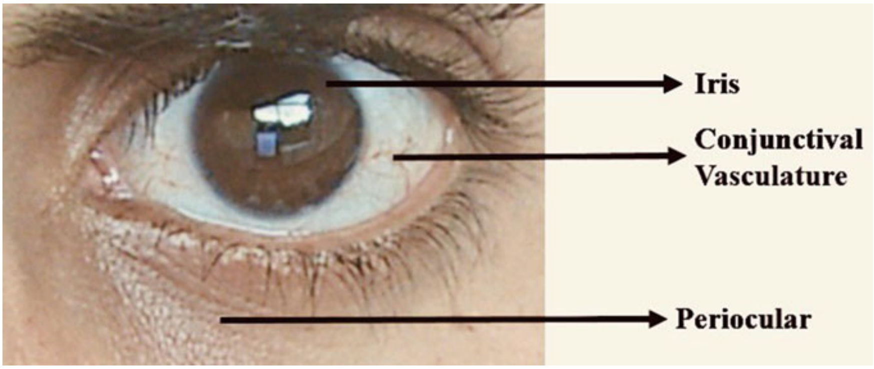


Fig. 1.7 Example eye image labeled with iris, conjunctival vasculature, and periocular region

LQP, and BSIF) and deep learning-based CNNs have been mostly used for identity verification in mobile ocular biometrics [23–26]. In 2016, a large-scale competition was conducted on Mobile Ocular Biometric Recognition on VISOB dataset [27]. Figure 1.6 shows substantial variations in the ocular images captured using the front-facing camera of iPhone 5s from publicly available VISOB mobile ocular biometrics dataset [27].

1.2.1.3 Fingerphoto

There has been a recent trend in touchless fingerprint recognition technology, where the back-facing smartphone cameras acquire high-resolution photographs of finger ridge patterns. This mobile modality is henceforth referred to as fingerphotos[10] [8, 28]. Fingerphoto authentication methods may offer an economical alternative to traditional fingerprint systems for mobile use cases as they avoid the need for extra hardware [29]. The further advantages of the touchless finger photo authentication methods over traditional touch-based fingerprint include being hygienic and removing the risk of leaving latent prints on the sensor. Furthermore, there are no finger impression deformations in the acquired images that could be caused by pressing the

[10] Though not a traditional selfie capture per se, and given its commonalities with selfie mobile biometrics, we have included it among other selfie modalities.

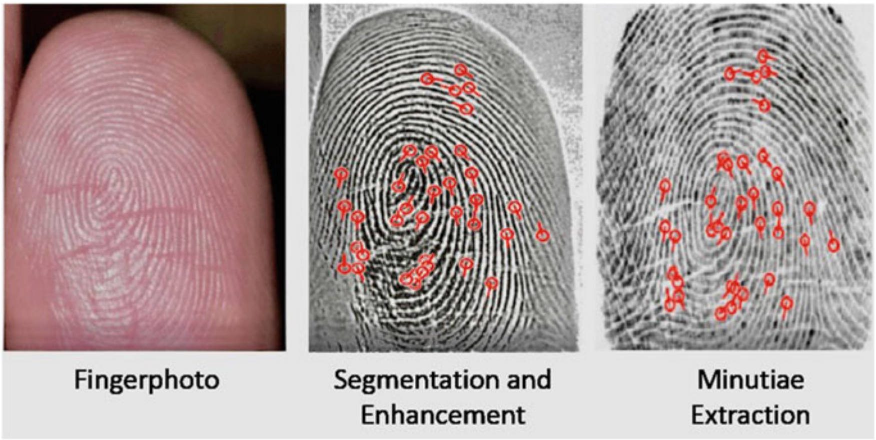


Fig. 1.8 Complete pipeline of fingerphoto-based system in mobile devices

finger on a touch-based sensor. Low-quality fingerprints due to low pressure or dry skin may also be mitigated by such touchless photographic fingerprint acquisition.

A typical pipeline for finger photo authentication system consists of imaging one or more fingers (with or without flash) with a high-quality back-facing mobile camera from a short distance. This is followed by image segmentation, enhancement, and minutiae extraction. The extracted minutiae form the template that is subsequently matched with an enrolled reference to establish the identity of the mobile user (see Fig. 1.8). Some of the finger photo challenges include the improper focus of the camera due to which ridge patterns of the finger may not be captured. Further, various potential poses of the finger must be considered: the orientation angle, pitch angle, and position of the finger, as well as the distance of the finger to the camera and the background.

1.2.2 Selfie Biometrics and Spoof Attacks

As the use of biometrics for smartphone user authentication continues to increase, capabilities to detect spoof attacks are needed to alleviate user concerns. A spoof attack occurs when an adversary mimics the biometric trait of another individual to circumvent the system for illegitimate access and advantages [30]. These attacks may pose a serious threat because they can be executed at the sensor (camera) level without requiring any technical knowledge of the functioning of the biometric system. Lack of efficient anti-spoofing and liveness detection methods may create a formidable psychological barrier in the mass adoption of biometrics in mobile applications. Therefore, there is a pressing need for the development of robust countermeasures against spoof attacks for mobile biometrics.

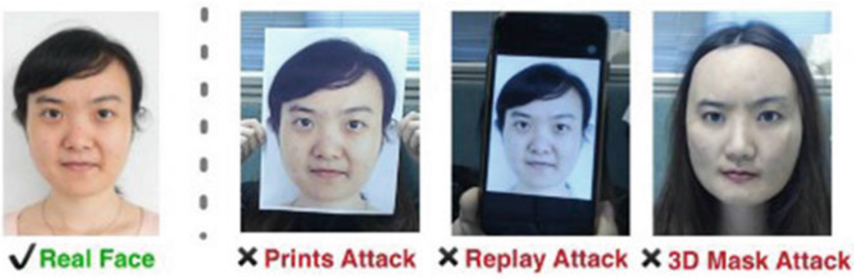


Fig. 1.9 Example of print, replay, and 3D mask attacks for face biometrics in mobile device [31]

In context of selfie biometrics, spoof attacks mainly consist of (i) *print attacks*, (ii) *photograph attacks*, and (iii) *replay attacks*. Print and photograph attacks can be executed using a selfie photograph of the enrolled user, which may be displayed in hard copy (2D or 3D) or on a screen to the mobile device. The video replay attacks are performed by displaying a video on a mobile screen. Anti-spoofing countermeasures aim to disambiguate live, and real face captures from spoof counterparts to avoid spoof attacks in mobile devices.

Face: Figure 1.9 shows example of print and replay attacks for face biometrics in the mobile device. Apart from print and photograph attacks, face recognition is also subject to 3D face mask attacks which require high-resolution fabrication system capturing the 3D shape and texture information of the target subject's face. However, print and replay attacks can be launched more easily by malicious users than 3D mask attacks.

The existing countermeasures can be coarsely classified into motion analysis-based [32–35], texture-based [36–44], image-quality based [39, 45–47], and deep learning-based (which can be considered as an end-to-end data-driven spoofing artifact feature extractor and classifier) [48]. Motion analysis-based methods can be considered as liveness detection, and texture, image quality and deep learning-based methods can be considered as spoof detection methods (since they mostly detect artifacts and distortions arising from spoofing methods).

Fingerphoto: The types of spoof attacks for finger photo can be photograph, print, and spoofs fabricated using material such as gelatin and latex. Countermeasures include use of textural descriptors such as local binary patterns, dense scale invariant feature transform, and locally uniform comparison image descriptor features combined with with classifiers such as support vector machine (SVM) [49], use of challenge response [50] and deep convolutional neural networks (which combines feature extraction and classification steps of the earlier mentioned methods) [51].

Ocular: Apart from photograph and print attacks, spoof attacks for ocular or iris biometrics may include the use of artificial eyes and patterned lenses. Common countermeasures include use of local and global textural descriptors such as LBP and GLCM [52], eye motion analysis, and convolutional neural networks, similar to face anti-spoofing [53].

1.2.3 Selfie and Cloud-Based Services

The significant challenge associated with selfie biometrics is the limited availability of resources—within the smartphone—for storage and computation. Therefore, it may be necessary in some cases to outsource the computing and storage demands to a more powerful server outside the smartphone. In this regard, cloud computing may be harnessed as a viable option [54, 55]. Cloud computing facilitates the outsourcing of computing and storage tasks to infrastructures managed by dedicated providers a potential approach to surpassing mobile resource limits. For instance, the feature extraction, data storage, and matching components of a biometric system can be moved to a cloud infrastructure, while leaving only the sensing task in the smartphone. There is an increased interest in performing biometric recognition in mobile devices and as a cloud-based service [54, 56, 57]. If the biometrics-in-the-cloud architecture is offered by a service provider, then it is referred to as Biometrics-as-a-Service (BaaS). If the infrastructure allows for component developers to develop and incorporate custom components in the cloud (e.g., feature extraction or matcher modules), then it is referred to as Platform-as-a-Service (PaaS). This paper in [54] presents a framework for Biometrics-as-a-Service (BaaS) that performs biometric matching operations in the cloud while relying on simple and ubiquitous consumer devices such as a smartphone.

1.2.4 Selfie and Soft Biometrics

Apart from biometric authentication using selfie images, several soft-biometric attributes can also be extracted from selfie captures. These soft-biometric attributes may include eyeglasses, gender, age, and clothing, which can be used in the absence of primary biometric trait, or conjunction with a primary biometric trait for performance enhancement. Also, these soft-biometric traits can also be used for continuous user authentication to verify that the user initially authenticated is still the user in control of the device [58]. Selfie soft biometrics including gender [59–61], age [62], eyeglasses [63], eyebrows [64], and clothing information [65] have been studied for use with mobile face and ocular modalities for performance enhancement and continuous user authentication (see Fig. 1.10). Further study in [66] proposed a combination of soft-biometric attributes such as face shape, skin tone, hair color, eyeglasses, ethnicity, and gender for continuous user authentication in mobile devices.

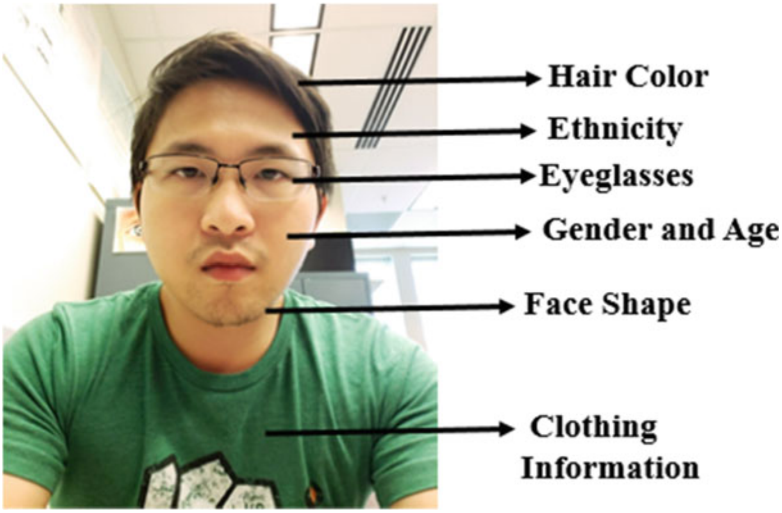


Fig. 1.10 Example of soft-biometric attributes from selfie images

1.3 Challenges and Future Directions

One of the main challenges in selfie biometrics involves developing accurate and computationally efficient methods for the mobile environment. Due to data acquisition in a mobile and uncontrolled environment, the acquired samples may exhibit substantial intra-class variations. This can lower the accuracy of the system and may even frustrate users of the devices.

A recent survey [21] suggests an average reported face recognition accuracy of 92.3% in a mobile environment. However, most of the existing methods are evaluated on in-house mobile datasets of limited size. Therefore, the relevance of the reported results cannot be established.

Reported error rates regarding the performance of proposed countermeasures against spoof attacks [21] in mobile devices are usually high, especially for replay attacks. This suggests the need for advanced and accurate methods for liveness and spoof detection for selfie biometrics. Continuous advancement in spoofing techniques will lead to novel methods for spoof attacks. There is an immediate need for designing a liveness detection/ anti-spoof method that is robust across new spoof attacks [67]. Therefore, the development of advanced and open-set liveness/ anti-spoof detection methods for known and novel spoof attacks should be the path forward.

With the advancement in mobile technology, deep learning-based solutions became viable for client-oriented and cloud-based mobile biometrics applications. Consequently, deep learning-based solutions for accurate recognition and anti-spoofing should be developed. Advanced loss functions such as triplet- [68] and center-loss [69] should be utilized for the task. There is a room for the development of a framework for Biometrics-as-a-Service that performs selfie matching operations in the cloud. Dynamic fusion framework needs to be developed for combining available soft-biometric attributes from selfie images for performance enhancement. Efforts should also be directed toward large-scale database collection for selfie face images to evaluate and compare deep learning solutions on a common test set.

1.4 Conclusion

Recently, several papers have been published on the topic of selfie biometrics. This book describes the state-of-the-art in selfie biometrics with a focus on the face, ocular, and finger modalities. This introductory chapter has described the notion of selfie biometrics and summarized the notable state of the art on this topic. This chapter will be followed by individual chapters covering: various selfie modalities, the methods of selfie-based mobile user authentication, predicting soft biometrics for performance enhancement and continuous authentication, anti-spoofing (measures and robustness), quality, privacy, security, and usability of selfie biometrics.

References

1. Jain A, Ross A, Nandakumar A (2011) Introduction to biometrics. Springer Publishers
2. Han S, Park H, Cho D, Park D, Lee S (2007) Face recognition based on near-infrared light using mobile phone. In: Beliczynski B, Dzielinski A, Iwanowski M, Ribeiro B (eds) Adaptive and natural computing algorithms, vol 4432. Lecture Notes in Computer Science. Springer, Heidelberg, pp 440–448
3. Jung S, Chung Y, Yoo J, Moon K (2008) Real-time face verification for mobile platforms. In: Bebis G, Boyle R, Parvin B, Koracin D, Remagnino P, Porikli F, Peters J, Klosowski J, Arns L, Chun Y, Rhyne T, Monroe L (eds) Advances in visual computing, vol 5359. Lecture Notes in Computer Science. Springer, Heidelberg, pp 823–832
4. Tao Q, Veldhuis R (2006) Biometric authentication for a mobile personal device. In: Third annual international conference on mobile and ubiquitous systems: networking services, San Jose, CA, pp 1–3
5. Walgamage T, Farook C (2014) A real-time hybrid approach for mobile face recognition. In: International conference on intelligent systems, modelling and simulation, pp 1–6
6. Rattani A, Derakhshani R (2017) Ocular biometrics in the visible spectrum: a survey. Image Vis Comput 59:1–16
7. Rattani A, Derakhshani R (2017) On fine-tuning convolutional neural networks for smartphone based ocular recognition. In: IEEE international joint conference on biometrics (IJCB), pp 762–767
8. Sankaran A, Malhotra A, Mittal A, Vatsa M, Singh R (2015) On smartphone camera based fingerphoto authentication. In: IEEE 7th international conference on biometrics theory, applications and systems, pp 1–7
9. Maiorana E, Campisi P, González-Carballo N, Neri A (2011) Keystroke dynamics authentication for mobile phones. In: ACM symposium on applied computing, New York, NY, USA, pp 21–26

10. Derawi MO, Nickel C, Bours P, Busch C (2010) Unobtrusive user-authentication on mobile phones using biometric gait recognition. In: Sixth international conference on intelligent information hiding and multimedia signal processing, pp 306–311
11. Tao Q, Veldhuis R (2010) Biometric authentication system on mobile personal devices. IEEE Trans Instrum Measur 59(4):763–773
12. Chen B, Shen J, Sun H (2012) A fast face recognition system on mobile phone. In: International conference on systems and informatics, Yantai, pp 1783–1786
13. Yang J, Chen X, Kunz W (2002) A PDA-based face recognition system. In: Sixth IEEE workshop on applications of computer vision, pp 19–23
14. Doukas C, Maglogiannis I (2010) A fast mobile face recognition system for android os based on eigenfaces decomposition. In: Papadopoulos H, Andreou A, Bramer M (eds) Artificial intelligence applications and innovations, vol 339. IFIP Advances in Information and Communication Technology. Springer, Heidelberg, pp 295–302
15. Kumar S, Singh P, Kumar V (2010) Architecture for mobile based face detection/recognition. Int J Comput Sci Eng 2(3):889–894
16. Yu H (2010) Face recognition for mobile phone using eigenfaces. University of Michigan, Tech. rep
17. Findling RD, Mayrhofer R (2012) Towards face unlock: on the difficulty of reliably detecting faces on mobile phones. In: International conference on advances in mobile computing and multimedia, Bali, Indonesia, pp 275–280
18. Kremic E, Subasi A, Hajdarevic K (2012) Face recognition implementation for client server mobile architecture. In: International conference on information technology interfaces, Dubrovnik, Croatia, pp 435–440
19. Mukherjee S, Chen Z, Gangopadhyay A, Russell A (2008) A secure face recognition system for mobile-devices without the need of decryption. In: Workshop on secure knowledge management, pp 11–16
20. Schneider C, Esau N, Kleinjohann L, Kleinjohann B (2006) Feature based face localization and recognition on mobile devices. In: International conference on control, automation, robotics and vision, Singapore, pp 1–6
21. Rattani A, Derakhshani R (2018) A survey of mobile face biometrics. Comput Electr Eng 72:39–52. https://doi.org/10.1016/j.compeleceng.2018.09.005, http://www.sciencedirect.com/science/article/pii/S004579061730650X
22. Gnther M, Costa-Pazo A, Ding C, Boutellaa E, Chiachia G, Zhang H, de Assis Angeloni M, Truc V, Khoury E, Vazquez-Fernandez E, Tao D, Bengherabi M, Cox D, Kiranyaz S, de Freitas Pereira T, Ganec Gros J, Argones-Ra E, Pinto N, Gabbouj M, Simes F, Dobriek S, Gonzlez-Jimnez D, Rocha A, Neto MU, Pavei N, Falco A, Violato R, Marcel S (2013) The 2013 face recognition evaluation in mobile environment. In: International conference on biometrics, Madrid, pp 1–7
23. Das A, Pal U, Ballester M, Blumenstein M (2014) A new efficient and adaptive sclera recognition system. In: IEEE symposium on computational intelligence in biometrics and identity management (CIBIM), pp 1–8
24. Park U, Ross A, Jain A (2009) Periocular biometrics in the visible spectrum: a feasibility study. In: IEEE 3rd international conference on biometrics: theory applications and systems, pp 1–6
25. Marsico MD, Nappi M, Proena H (2017) Results from miche ii mobile iris challenge evaluation ii, Pattern Recogn Lett 91(C):3–10
26. Reddy N, Rattani A, Derakhshani R (2018) Ocularnet: deep patch-based ocular biometric recognition. In: 2018 IEEE international symposium on technologies for homeland security (HST), pp 1–6. https://doi.org/10.1109/THS.2018.8574156
27. Rattani A, Derakhshani R, Saripalle SK, Gottemukkula V (2016) ICIP 2016 competition on mobile ocular biometric recognition. In: IEEE International Conference on image processing, challenge session on mobile ocular biometric recognition, Phoenix, AZ, pp 320–324
28. Stein C, Nickel C, Busch C (2012) Fingerphoto recognition with smartphone cameras. In: BIOSIG—Proceedings of the international conference of biometrics special interest group, pp 1–12

29. Carney LA, Kane J, Mather JF, Othman A, Simpson AG, Tavanai A, Tyson RA, Xue Y (2017) A multi-finger touchless fingerprinting system: mobile fingerphoto and legacy database interoperability. In: Proceedings of the 2017 4th international conference on biomedical and bioinformatics engineering, ICBBE 2017, New York, NY, USA, pp 139–147
30. Chingovska I, dos Anjos AR, Marcel S (2014) Biometrics evaluation under spoofing attacks. IEEE Trans Inf Forensics Secur 9(12):2264–2276
31. Liu S, Yang B, Yuen P, Zhao G (2016) A 3D mask face anti-spoofing database with real world variations. In: The IEEE conference on computer vision and pattern recognition (CVPR) workshops, pp 1551–1557
32. Patel K, Han H, Jain AK (2016) Cross-database face antispoofing with robust feature representation. In: You Z, Zhou J, Wang Y, Sun Z, Shan S, Zheng W, Feng J, Zhao Q (eds) Biometric recognition. Springer International Publishing, Cham, pp 611–619
33. Siddiqui IA, Bharadwaj S, Dhamecha TI, Agarwal A, Vatsa M, Singh R, Ratha N (2016) Face anti-spoofing with multifeature videolet aggregation. In: International conference on pattern recognition, Cancun, pp 1035–1040
34. Tirunagari S, Poh N, Windridge D, Iorliam A, Suki N, Ho ATS (2015) Detection of face spoofing using visual dynamics. IEEE Trans Inf Forensics Secur 10(4):762–777
35. Pinto A, Pedrini H, Schwartz WR, Rocha A (2015) Face spoofing detection through visual codebooks of spectral temporal cubes. IEEE Trans Image Process 24(12):4726–4740
36. Akhtar Z, Michelon C, Foresti GL (2014) Liveness detection for biometric authentication in mobile applications. In: 2014 international Carnahan conference on security technology, Rome, pp 1–6
37. Chingovska I, Anjos A, Marcel S (2012) On the effectiveness of local binary patterns in face anti-spoofing. In: International conference of biometrics special interest group (BIOSIG), Germany, pp 1–7
38. Boulkenafet Z, Komulainen J, Li L, Feng X, Hadid A (2017) OULU-NPU: a mobile face presentation attack database with real-world variations. In: IEEE international conference on automatic face gesture recognition, Washington, DC, pp 612–618
39. Costa-Pazo A, Bhattacharjee S, Vazquez-Fernandez E, Marcel S (2016) The replay-mobile face presentation-attack database. In: International conference of the biometrics special interest group, Germany, pp 1–7
40. Boulkenafet Z, Komulainen J, Hadid A (2016) Face spoofing detection using colour texture analysis. IEEE Trans Inf Forensics Secur 11(8):1818–1830
41. Arashloo SR, Kittler J, Christmas W (2015) Face spoofing detection based on multiple descriptor fusion using multiscale dynamic binarized statistical image features. IEEE Trans Inf Forensics Secur 10(11):2396–2407
42. Gan J, Li S, Zhai Y, Liu C (2017) 3D convolutional neural network based on face anti-spoofing. In: International conference on multimedia and image processing, Wuhan, pp 1–5
43. Atoum Y, Liu Y, Jourabloo A, Liu X (2017) Face anti-spoofing using patch and depth-based CNNs. In: IEEE international joint conference on biometrics, Denver, CO, pp 319–328
44. Pereira F, Komulainen J, Anjos A, Martino MD, Hadid A, Pietikäinen M, Marcel S (2014) Face liveness detection using dynamic texture. EURASIP J Image Video Process 2014(1):2
45. Patel K, Han H, Jain AK, Ott G (2015) Live face video vs. spoof face video: use of moire patterns to detect replay video attacks. In: International conference on biometrics, Phuket, pp 98–105
46. Wen D, Han H, Jain AK (2015) Face spoof detection with image distortion analysis. IEEE Trans Inf Forensics Secur 10(4):746–761
47. Galbally J, Marcel S (2014) Face anti-spoofing based on general image quality assessment. In: International conference on pattern recognition, Stockholm, pp 1173–1178
48. Boulkenafet Z, Komulainen J, Akhtar Z, Benlamoudi A, Samai D, Bekhouche SE, Ouafi A, Dornaika F, Taleb-Ahmed A, Qin L, Peng F, Zhang LB, Long M, Bhilare S, Kanhangad V, Costa-Pazo A, Vazquez-Fernandez E, Perez-Cabo D, Moreira-Perez JJ, Gonzalez-Jimenez D, Mohammadi A, Bhattacharjee S, Marcel S, Volkova S, Tang Y, Abe N, Li L, Feng X, Xia Z, Jiang X, Liu S, Shao R, Yuen PC, Almeida WR, Andalo F, Padilha R, Bertocco G, Dias W,

Wainer J, Torres R, Rocha A, Angeloni MA, Folego G, Godoy A, Hadid A (2017) A competition on generalized software-based face presentation attack detection in mobile scenarios. In: IEEE international joint conference on biometrics, Denver, CO, pp 688–696

49. Taneja A, Tayal A, Malhorta A, Sankaran A, Vatsa M, Singh R (2016) Fingerphoto spoofing in mobile devices: a preliminary study. In: IEEE international conference on biometrics theory, applications and systems, pp 1–7
50. Stein C, Bouatou V, Busch C (2013) Video-based fingerphoto recognition with anti-spoofing techniques with smartphone cameras. In: International conference of the BIOSIG Special Interest Group (BIOSIG), pp 1–12
51. Fujio M, Kaga Y, MurakamiT, Ohki T, Takahashi K (2018) Face/fingerphoto spoof detection under noisy conditions by using deep convolutional neural network. In: International joint conference on biomedical engineering systems and technologies, pp 54–62
52. Sequeira AF, Murari J, Cardoso JS (2014) Iris liveness detection methods in the mobile biometrics scenario. In: International joint conference on neural networks (IJCNN), pp 3002–3008
53. Sequeira AF, Oliveira HP, Monteiro JC, Monteiro JP, Cardoso JS (2014) Mobilive 2014 mobile iris liveness detection competition. In: IEEE international joint conference on biometrics, pp 1–6
54. Talreja V, Ferrett T, Valenti MC, Ross A (2018) Biometrics-as-a-service: a framework to promote innovative biometric recognition in the cloud. In: IEEE international conference on consumer electronics (ICCE), pp 1–6
55. Mell P, Granc T (2011) The nist definition of cloud computing. Tech. rep, Recommendations of the National Institute of Standards and Technology
56. Chow R, Jakobsson M, Masuoka R, Molina J, Niu Y, Shi E, Song Z (2010) Authentication in the clouds: a framework and its application to mobile users. In: ACM cloud computing security workshop (CCSW), New York, NY, USA, pp 1–6
57. Barra S, Casanova A, Narducci F, Ricciardi S (2015) Ubiquitous iris recognition by means of mobile devices. Pattern Recogn Lett 57:66–73
58. Patel VM, Chellappa R, Chandra D, Barbello B (2016) Continuous user authentication on mobile devices: recent progress and remaining challenges. IEEE Signal Process Mag 33(4):49–61
59. Rattani A, Reddy N, Derakhshani R (2017) Gender prediction from mobile ocular images: a feasibility study. In: IEEE international symposium on technologies for homeland security, pp 1–6
60. Buriro A, Akhtar Z, Crispo B, Frari FD (2016) Age, gender and operating-hand estimation on smart mobile devices. In: International conference of the biometrics special interest group, pp 1–5
61. Rattani A, Reddy N, Derakhshani R (2018) Convolutional neural networks for gender prediction from smartphone-based ocular images. IET Biometrics 7:423–430
62. Rattani A, Reddy N, Derakhshani R (2017) Convolutional neural network for age classification from smart-phone based ocular images. In: 2017 IEEE international joint conference on biometrics (IJCB), pp 756–761. https://doi.org/10.1109/BTAS.2017.8272766
63. Mohammad AS, Rattani A, Derahkshani R (2017) Eyeglasses detection based on learning and non-learning based classification schemes. In: IEEE international symposium on technologies for homeland security (HST), pp 1–5. https://doi.org/10.1109/THS.2017.7943484
64. Mohammad AS, Rattani A, Derakhshani R (2018) Short-term user authentication using eyebrows biometric for smartphone devices. In: IEEE computer science and electronic engineering conference, pp 1 – 6
65. Nguyen H, Sai R, Li Z, Derakhshan R (2018) User re-identification using clothing information for smartphones. In: IEEE international symposium on technologies for homeland security (HST), pp 1–5
66. Samangouei P, Patel VM, Chellappa R (2015) Attribute-based continuous user authentication on mobile devices. In: IEEE 7th international conference on biometrics theory, applications and systems (BTAS), pp 1–8

67. Rattani A, Scheirer WJ, Ross A (2015) Open set fingerprint spoof detection across novel fabrication materials. IEEE Trans Inf Forensics Secur 10(11):2447–2460
68. Schroff F, Kalenichenko D, Philbin J. FaceNet: a unified embedding for face recognition and clustering, CoRR abs/1503.03832
69. Wen Y, Zhang K, Li Z, Qiao Y (2016) A discriminative feature learning approach for deep face recognition. In: Leibe B, Matas J, Sebe N, Welling M (eds) European conference on computer vision. Cham, pp 499–515

Part I
Selfie Finger, Ocular and Face Biometrics

Chapter 2
User Authentication via Finger-Selfies

Aakarsh Malhotra, Shaan Chopra, Mayank Vatsa and Richa Singh

Abstract In the last one decade, the usage and capabilities of smartphones have increased multifold. To keep data and devices secure, fingerprint and face recognition-based unlocking are gaining popularity. However, the additional cost of installing fingerprint sensors on smartphones questions the use of fingerprints. Alternatively, finger-selfie, an image of a person's finger acquired using a built-in smartphone camera, can act as a cost-effective solution. Unlike capturing face selfies, capturing good-quality finger-selfies may not be a trivial task. The captured finger-selfie might incorporate several challenges such as illumination, in- and out-of-plane rotations, blur, and occlusion. Users may even present multiple fingers together in the same frame. In this chapter, we propose authentication using finger-selfies taken in an unconstrained environment. The research contributions include the UNconstrained FIngerphoTo (UNFIT) database which is captured under challenging unconstrained conditions. The database also contains the manual annotation of identities and location of the fingers. We further present a segmentation algorithm to segment finger regions and, finally, perform feature extraction and matching using CompCode and ResNet50. Experimental results show that despite multiple challenges present in the UNFIT database, the segmentation algorithm can segment and perform authentication using finger-selfies.

Aakarsh Malhotra and Shaan Chopra: Equal contribution by student authors.

A. Malhotra · S. Chopra · M. Vatsa · R. Singh (✉)
IIIT-Delhi, Delhi, India
e-mail: rsingh@iiitd.ac.in

A. Malhotra
e-mail: aakarshm@iiitd.ac.in

S. Chopra
e-mail: shaan15090@iiitd.ac.in

M. Vatsa
e-mail: mayank@iiitd.ac.in

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision
and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_2

2.1 Introduction

In the current digital era, smartphones and mobile devices are ubiquitous. With the growth of smartphone usage, people store enormous amounts of personal and confidential information on their smartphones. Storing such information on smartphones demands suitable security mechanisms. Traditional security measures include passwords, patterns, or pins. However, these methods need to be memorized by the users and are vulnerable to shoulder surfing attacks [1]. Alternatively, biometric-based user authentication is now more popular and requires minimal effort from the users.

As illustrated in Fig. 2.1, modern smartphones have multiple sensors that can facilitate user authentication. For instance, cameras can be used to capture face [2] and finger-selfies, while fingerprint sensors can be used to acquire fingerprints. Researchers and commercial entities have explored the usability of all three, and each posing certain advantages and constraints. For instance, traditional fingerprints are accurate but require the installation of additional capacitive sensors [3]. Face selfies are easy to capture, but they may be affected by several external factors. Similarly, finger-selfies do not need any additional sensors, but the technology requires more

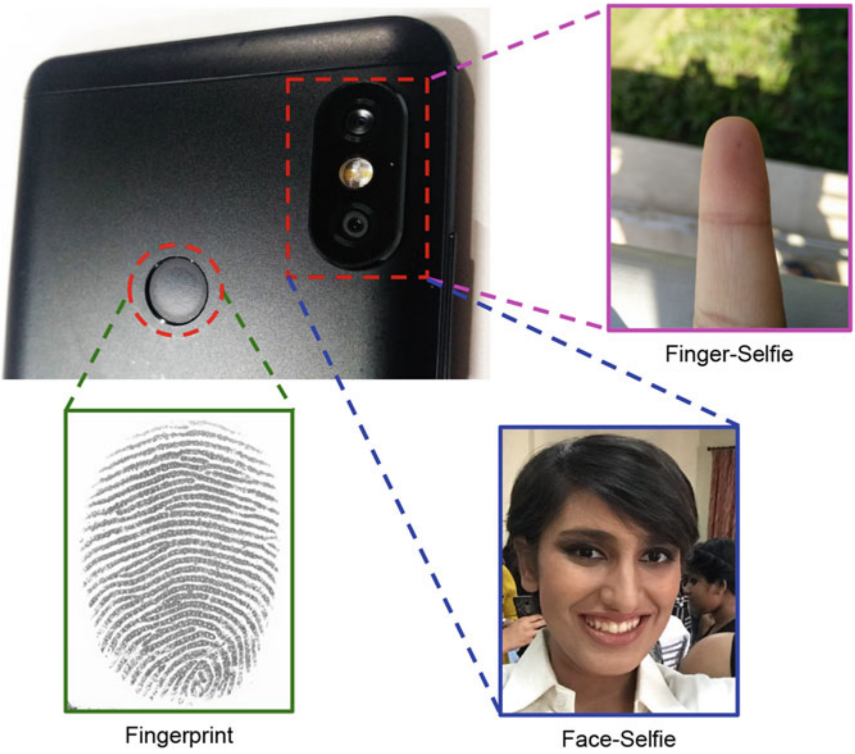


Fig. 2.1 Acquisition sensors and their corresponding captured modalities

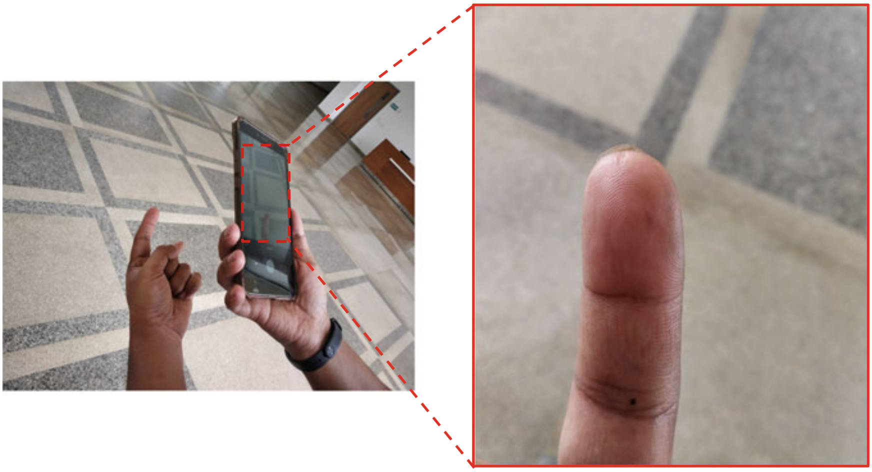


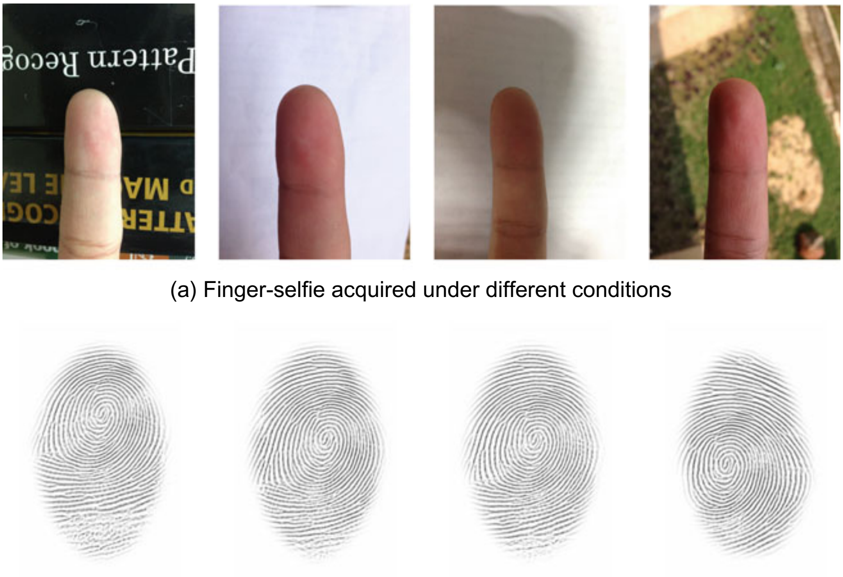
Fig. 2.2 An illustration of acquisition mechanism of finger-selfie and the corresponding finger-selfie

research to demonstrate the effectiveness. This chapter focuses on finger-selfie, presenting a review of the research efforts related to improving the usability and accuracy of finger-selfie recognition.

As shown in Fig. 2.2, finger-selfie acquisition involves capturing ridge-valley details present on the tip of the finger using a device camera by the user. Overcoming the drawback of traditional biometric-based authentication, a finger-selfie does not require an additional sensor. All it needs is the smartphone's in-built camera. As per Tim Ahonen's Phone book [4] and Statista [5], approximately 89% of all digital photographs arise from handheld devices such as tablets and smartphones. While these statistics motivate the use of finger-selfies as a cost-effective method for authentication, there are other advantages as well. Finger-selfies act as a contactless fingerprint acquisition technique, which is hygienic and secure, leaving no latent impressions on the surface of the sensor. Over the flattened live scan fingerprints, finger-selfies also contain additional information such as finger shape and phalanx lines. While these lines may not have global uniqueness, a localized correlation with ridge-valley patterns in the neighborhood may aid person identification [6].

Other than authentication for device unlocking, law enforcement agencies have also shown their interest toward finger-selfies. For instance, on finding a finger-selfie of a potential drug dealer holding drugs on his fingers, the South Wales Police and the scientific support unit utilized the finger-selfie to identify the culprit [7]. Similarly, a hacker used an image of a German minister's finger, acquired from a distance of three meters, to generate fingerprints [8]. Such use cases highlight the need for finger-selfie-based recognition systems.

Emphasizing on the other side of the coin, finger-selfie-based user authentication is not perfect either. As illustrated in Fig. 2.3, a finger-selfie looks drastically different from a traditional fingerprint, with skin and background visible along with ridge-



(a) Finger-selfie acquired under different conditions

(b) Corresponding livescan images of the same subject

Fig. 2.3 Visual difference between a finger-selfie and a legacy fingerprint image

valley details. While its acquisition requires minimal effort from the user, their lack of cooperation might induce many challenges. Unlike capturing face selfies, acquiring a good-quality finger-selfie may not be a trivial task, and the captured finger-selfie might comprise several variations such as illumination, in- and out-of-plane rotations, blur, and occlusion. Users might even present multiple fingers in the same frame. A summary of these challenges is illustrated in Fig. 2.4. While these challenges highlight a real-life unconstrained acquisition scenario, detection and recognition of these finger-selfies for smartphone authentication become a cumbersome task.

To promote unconstrained finger-selfie-based recognition, this chapter first provides a review of existing research on finger-selfie followed by finger-selfie-based authentication in an unconstrained environment. This research is inspired by our preliminary work, which showcased the application of finger-selfies in an unconstrained environment [9]. The important research contributions of this chapter are:

1. A review of existing databases utilized in the literature for finger-selfie/image/photograph-based recognition and a detailed summary of existing approaches for finger-selfie recognition are discussed.
2. A novel publicly available UNconstrained FIngerphoTo (UNFIT) database, which is captured under challenging unconstrained conditions. The database also contains manual annotation of identities and location for 3450 images from 115 subjects.

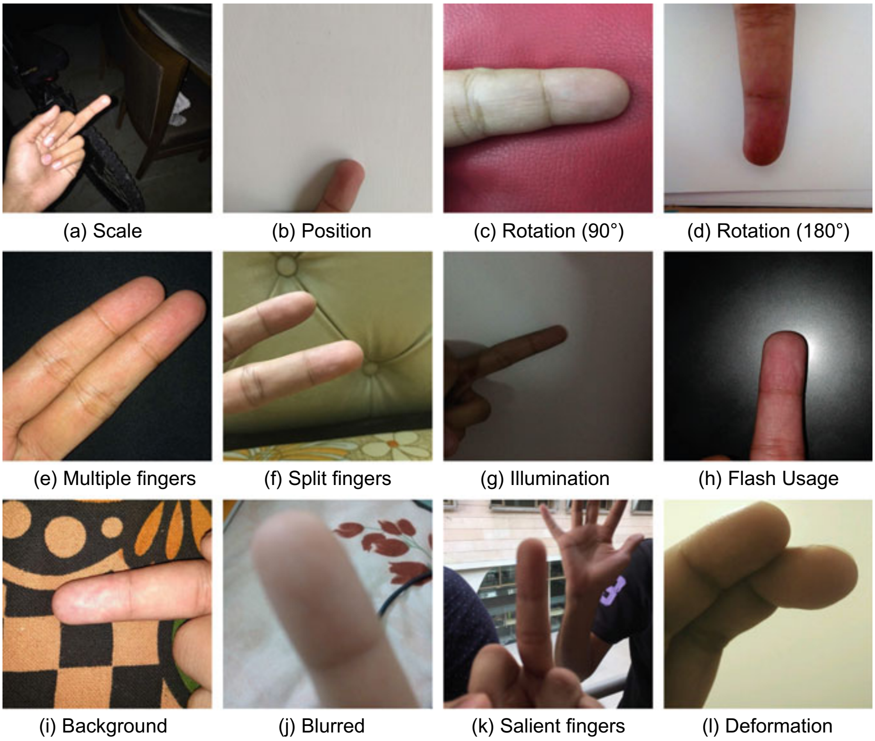


Fig. 2.4 Sample finger-selfie images from the proposed UNFIT database. While the database incorporates numerous challenges, a real-life unconstrained acquisition of finger-selfies might contain one or more challenges together, making finger-selfie recognition a complex problem. Varying resolutions of the camera adds to the challenges of finger-selfie recognition

3. A segmentation algorithm to segment finger regions from a finger-selfie using the existing VGG SegNet [10] model. The performance of the segmentation algorithm is compared with other segmentation methods such as FCN 8 [11]. We show that existing deep learning algorithms for segmentation can easily outperform the traditional skin color-based segmentation [12] methods used in the literature.
4. Finally, recognition of the segmented finger is performed. The benchmarking for feature extraction and matching is performed using CompCode [13] and ResNet50 [14] followed by Hamming distance and cosine similarity, respectively. Experimental results show that despite multiple challenges present in the UNFIT database, finger-selfie-based biometric authentication is feasible and pragmatic.

2.2 Related Work

Recent studies have demonstrated the usage of fingerphoto/contactless fingerprints acquired using smartphones and other digital cameras toward benchmarking of contactless fingerprint recognition. However, a significant limitation of these studies is the use of constrained or semi-constrained fingerphoto datasets. A summary of the datasets is presented in Table 2.1, and their details are given below.

Table 2.1 Literature review of existing databases of contactless fingerprints/fingerphotos

Research	Device	Subjects	# Samples	Challenges	Public	Nature
Song et al. [15]	CCD	–	–	None	✗	Constrained
Lee et al. [16]	Phone	150 + 168	400 + 840	Background, orientation	✗	Semi-constrained
Lee et al. [17]	Phone	15	60 + 30 + 30 videos	Blur, orientation/ movement	✗	Semi-constrained
Piuri and Scotti [18]	Webcam	15	150	Background	✗	Semi-constrained
Hiew et.al [19]	Digital Camera	103 classes	1938	None	✗	Constrained
Kumar and Zhou [6]	Webcam	156	1566	Resolution	✓	Semi-constrained
Derawi et al. [20]	Phone	22	1320	None	✗	Constrained
Yang et al. [21–23]	Phone	25	2100	Background, illumination	✗	Semi-constrained
Stein et al. [24]	Phone	11 + 37	66 videos, 990 photographs	None	✗	Constrained
Tiwari and Gupta [25]	Phone	50	150	Illumination	✗	Constrained
Sankaran et al. [12]	Phone	64	4096	Background, illumination	✓	Semi-constrained
Taneja et al. [26]	Multiple	64	8192	Fingerphoto Spoofing	✓	–
Lin and Kumar [27]	-	300 classes	1800	None	✓	Constrained
Proposed	**Phone**	**230 classes**	**3450**	Background, blur, multiple fingers, illumination, affine variation, resolution, deformations	✓	**Unconstrained**

2.2.1 Existing Databases

Several researchers have designed algorithms and shown results on contactless fingerprint recognition. However, a significant limitation related to the research on finger-selfie recognition is the unavailability of public datasets. While four of the datasets are publicly available, these datasets incur just one or two variations, which lack common challenging scenarios of acquisition present in finger-selfies. A summary of these datasets is presented below.

2.2.1.1 Publicly Available Databases

As illustrated in Table 2.1, there exist databases for contactless fingerprints; however, for benchmarking and algorithmic evaluation, only the following databases are publicly available in the research community:

- HKPU Low-Resolution Fingerprint Database [6]: The database has a total of 1566 low-resolution contactless fingerprint images from 156 subjects. The contactless fingerprints are acquired using a webcam in two different sessions. While the database is acquired at a low resolution, it incorporates no other challenge during acquisition. Hence, the database can be termed as semi-constrained.
- IIITD Smartphone Fingerphoto Database [12]: In 2015, Sankaran et al. proposed this database, containing 4096 fingerphoto images from 64 participants acquired using a smartphone camera. The database also includes 1024 livescan images to promote matching of fingerphoto with legacy fingerprint databases. The subsets of the database include varying background and illumination. Hence, this database can also be considered as semi-constrained.
- PolyU Contactless to Contact-based Fingerprint Database [27]: Recently, Lin and Kumar proposed a constrained dataset, with 1800 contactless fingerprint samples from 300 different fingers. While the images of fingers were acquired in a constrained setting, the database aimed to establish the matching of contactless fingerprints with contact-based livescan fingerprints. Hence, the database also includes 1800 contact-based livescan images.
- Other than the databases mentioned above, Taneja et al. [26] proposed a Spoofed Fingerphoto Database, which aimed to establish the effect of spoofing of fingerphotos using display and print attack. This database was created using fingerphotos taken from the IIITD Smartphone Fingerphoto Database [12].

Using the in-house and publicly available touchless fingerprint databases, researchers have demonstrated benchmarking results of their proposed algorithms. A summary of these algorithms is presented below.

2.2.2 *Finger-Selfie Recognition Techniques*

For touchless fingerprint recognition, Song et al. [15] used only blue channel information of finger images. They utilized mean and coherence for segmentation and Gabor filters to enhance ridge details. Their results were illustrated visually on a touchless fingerprint image. In 2006, Lee et al. [16] performed segmentation by combining normalized color (RB) model and frequency information extracted using the Tenengrad method. Minutiae were extracted from the segmented image, following which the authors reported about 80% GAR at 0.01% FAR. In 2008, Lee et al. [17] aimed at focus estimation by estimating blur. They also used coherence and symmetry for quality estimation and difference in frames (contour extraction) for pose estimation. On the Samsung Database (SDB)—I, II, III, IV—with 60, 30, 30 image sequences and 1200 fingerprint images, respectively, authors reported a rejection rate of 5.67% and EER of 3.02%.

Piuri and Scotti [18] performed blur reduction using Lucy-Richardson algorithm and Wiener filter algorithm followed by color model and morphology-based segmentation. After performing fingerphoto registration, enhancement, and minutia extraction using MINDTCT, authors reported an EER of 0.042% for 150 images. Hiew et al. [19] utilized Gabor features, followed by PCA and SVM for verification. They reported an EER of 1.23%. In 2011, while proposing a publicly available dataset, Kumar and Zhou [6] performed enhancement by Sobel filtering and area thresholding on the acquired image, followed by Gaussian sharpening. Using LRT and CompCode features followed by Hamming distance, the authors reported a cross-session EER of 3.95% with 93.97% accuracy on the proposed dataset. In the same year, Derawi et al. [20] performed feature extraction and matching using COTS and reported an EER of 0.00–23.62% for different fingers on their in-house database.

Yang et al. [21–23] utilized their semi-constrained database with 2100 samples toward quality assessment of fingerprint images captured from a smartphone camera. They defined a total of seven [21] and twelve [22] quality metrics to determine the quality of contactless fingerprint image. Using the same dataset, Raghavendra et al. [23] performed mean shift clustering to segment the probable finger regions. The final finger is detected from top five-sized regions using a fusion of Pearson, Fourier magnitude, and energy measure based on the wavelet transform. They reported an average segmentation accuracy of 96.46%. Using NBIS MINDTCT for minutia extraction followed by matching, authors report an EER of 3.74%. In 2013, Stein et al. [24] performed spoof detection, followed by minutia extraction and matching. The authors reported 1.20% EER for contactless fingerprints and 3.00% EER for finger videos. Tiwari and Gupta [25] found ROI in fingerphoto by adaptive thresholding followed by morphological operations. They aligned the image using PCA followed by image enhancement using adaptive histogram equalization. Using SURF features, authors report an EER of 3.33% on their proposed in-house database.

In 2015, Sankaran et al. [12] created IIITD Smartphone Fingerphoto Database and proposed a fingerphoto-to-fingerphoto and fingerphoto-to-livescan matching algorithm. With segmentation performed using adaptive thresholding, authors performed image sharpening and median filtering to enhance the image [28]. From the enhanced image, ScatNet features were extracted, followed by PCA and matching using RDF classifier. On the proposed semi-constrained dataset, authors reported an EER of 3.65–7.45% on different subsets of fingerphoto-to-fingerphoto matching and 7.07–10.43% for fingerphoto-to-livescan matching. Later, in 2017, Malhotra et al. [29] further improved the state-of-the-art performance on IIITD Smartphone Fingerphoto Database. Using an LBP-based enhancement, the authors reported an EER of 1.47–8.36% on different subsets of fingerphoto-to-fingerphoto matching and 6.44–7.61% for fingerphoto-to-livescan matching. Recently, Lin and Kumar [27] proposed a livescan and contactless fingerprint image database. To align the contactless images with livescan images, the authors proposed an RTPS-based fingerprint deformation correction model. By performing minutiae- and ridge-based matching, the authors reported a rank-1 accuracy of 94.11% using their proposed algorithm.

While these algorithms have shown good accuracies and low error rates, their performance is not evaluated in a real-life scenario of unconstrained finger-selfie recognition. A primary reason is the absence of an unconstrained finger-selfie database. To address this concern and to promote finger-selfie recognition in an uncontrolled scenario, we present UNFIT: an unconstrained fingerphoto database in the next section.

2.3 UNconstrained FingerPhoto (UNFIT) Dataset

In Sect. 2.2.1.1, we highlighted publicly available databases for contactless fingerprint recognition. While these datasets have an ample number of samples, these samples are acquired in a constrained or semi-constrained environment. In this research, we create the first unconstrained fingerphoto (UNFIT) database and make it available for the research community.[1] The database has many challenges, which would be present in a finger-selfie acquired in an uncontrolled environment with minimal user cooperation. The details of the dataset are presented below.

2.3.1 Database Acquisition

Forty-five different smartphones belonging to the subjects are used to capture finger-selfies. This brings variations in terms of resolution and camera sensor to the database. OnePlus and iPhone devices are used to acquire 48% of images in the database followed by other phones including Redmi devices, Google Nexus, Lenovo K3 Note, Lenovo K4, Mi 4, Le 1s, Samsung Galaxy, Micromax Canvas, Moto G, Moto C, Moto M, and HTC devices. The camera resolutions of these smartphones varied from 8 to 16 MP. The distribution of different smartphone devices used for finger-selfie acquisition can be seen in Fig. 2.5a.

[1] The UNFIT database can be downloaded from: http://iab-rubric.org/resources/UNFIT.html.

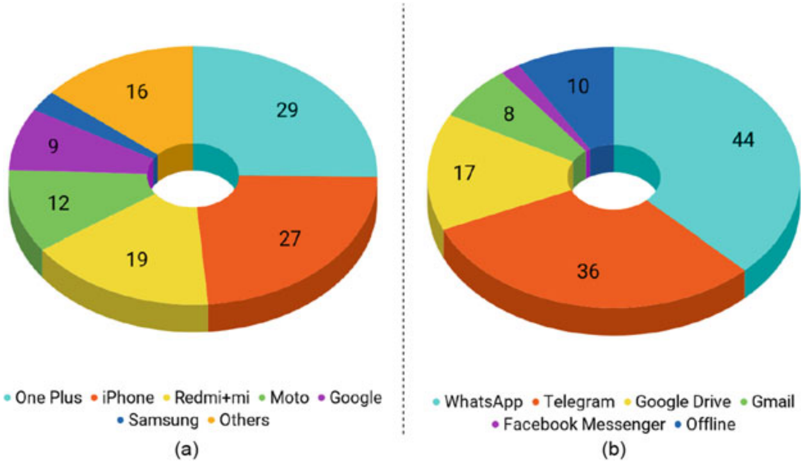


Fig. 2.5 Acquisition details: **a** Devices used for finger-selfie acquisition, and **b** Offline and online mechanisms used for obtaining finger-selfies

The database is collected via both online and offline methods which helps incorporate the effect of image compression due to transmission. WhatsApp, Telegram, Google Drive, Gmail, and Facebook messenger are used for online data collection, whereas for offline data collection, different phone devices belonging to the subjects are used followed by transmission via a pen drive. Figure 2.5b shows the distribution of images collected using different modes of online and offline data collection. Adding on, variations in illumination, intensity, and blur are present in the database due to the optional usage of auto-focus and flash for acquiring finger-selfies.

During database acquisition, no constraints are enforced for distance of the finger from the camera sensor. Varying distance allows the presence of more challenges, such as position and scale variation. However, the appearance of ridge-valley details stays limited with respect to the camera sensor. The minimum and maximum distances for a focussed detailed acquisition depend upon the camera's aperture and len's focal length. With 45 different smartphones used to obtain finger-selfies, the camera's aperture and len's focal length vary across the smartphone devices. Hence, a generic claim for a minimum and maximum distance for a focussed image cannot be made. Thus, varying sensors, lens, the distance of finger, illumination, and background variations makes locating, segmenting, and recognizing ridge-valley details in the finger challenging.

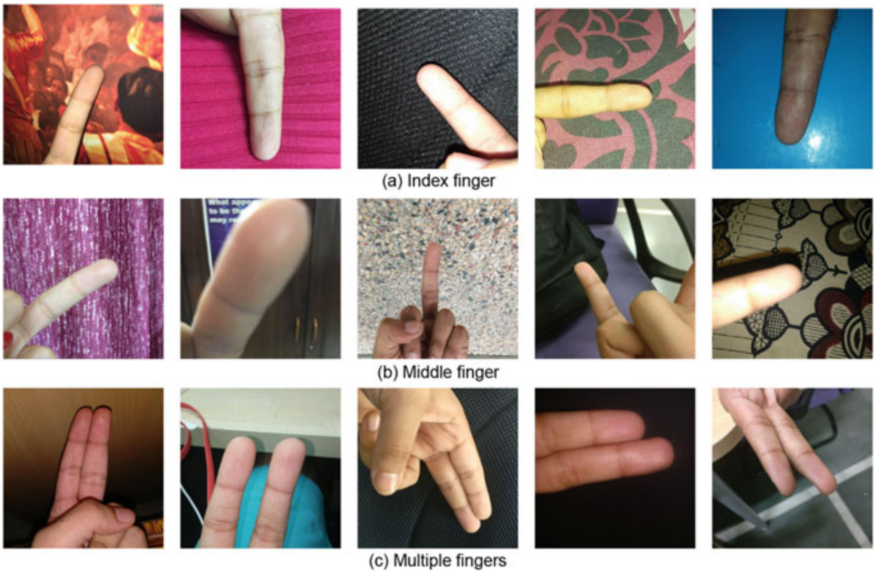
2.3.2 Database Statistics

Over a span of three months, we collated a novel finger-selfie database consisting of 3450 images and termed it as Unconstrained FIngerphoTo (UNFIT) database. The database has multiple images of the index and the middle finger for each subject, where both the fingers of the same participants are considered as different classes. We refrained from acquiring thumb finger-selfies since capturing frontal region of thumb while holding a phone facing downward in the other hand is inconvenient for subjects. During acquisition, the participants are allowed to use either of the hand for capturing the finger-selfies, as long as all the finger-selfies arise from the same hand. The database contains 230 different classes belonging to 115 participants. Out of the 115 subjects from whom finger-selfies were captured in the UNFIT database, 38 were female participants, and 77 were male participants. The details of the database can be seen in Table 2.2. Figure 2.6 exhibits some sample images from the database. Two different sets of finger-selfies are collected from each subject:

- **Set I: Single Finger**—Images of the index and middle fingers belonging to the same hand of a user are captured. Finger-selfies are collected from either the left

Table 2.2 A summary of various subsets presents in the UNFIT database

Subset	Fingers	Classes	Images
Set I	Index	115	1150
	Middle	115	1150
Subtotal:		230	2300
Set II	Multiple fingers	115	1150
Total:			3450


(a) Index finger

(b) Middle finger

(c) Multiple fingers

Fig. 2.6 Sample finger-selfie images from different subsets of the proposed UNFIT database

or right hand of the user as per his/her convenience without enforcement of any constraints regarding background, illumination, resolution, position, or orientation of the finger. Figure 2.6a and b demonstrates sample images belonging to this set. The set contains a total of 2300 images (=115 subjects × 2 fingers × 10 instances per finger).

- **Set II: Multiple Fingers**—At times, users may capture multiple fingers, intentionally or unintentionally, and this additional information can be useful for improving finger-selfie recognition performance. Thus, this is useful for demonstrating the effect of multiple fingers on finger-selfie recognition. Figure 2.6c shows the sample images belonging to this set. The set contains a total of 1150 samples (=115 subjects × 10 instances per participant) of both index and middle fingers belonging to the same hand taken together.

2.3.3 Challenges

In a scenario where the user cooperation is minimal, intra-class variations may increase. Some of these variations are shown in Fig. 2.4. A detailed description of challenges included in the proposed UNFIT database is as follows:

- **Affine variations**: Finger-selfie acquisition involves presenting the finger in front of the rear or front camera of the smartphone. While this task sounds trivial, there can be enormous affine variations. These variations may include translation and rotation of finger. Rotation variation may be caused both by rotation of finger in the 2D image plane (Fig. 2.4c–d) and by rolling of the finger on axis of the finger. While rotation in the 2D image plane does not lead to any information loss, a rotation along the finger axis may result in different amount of acquired ridge-valley detail. The varying distance from the acquisition camera would result in scale variations.
- **Multiple fingers**: As a part of the UNFIT dataset, index and middle fingers are collected together. While the multiple fingers can be placed in any order and may experience all variations a single finger can, multiple fingers may encounter other challenges as well. As illustrated in Fig. 2.4e–f, the multiple fingers may be split or may be presented together. The split-finger scenario aids in the robust testing of segmentation algorithms, since the algorithms should be able to segment the fingers in both situations.
- **Illumination**: The finger-selfies can be captured in both indoor and outdoor environments. It induces illumination variations, which may result in dull or bright finger-selfies. Usage of camera flash, as illustrated in Fig. 2.4h, may result in targeted bright regions too.
- **Background**: Allowing any natural background to be present, finger-selfies may have similar looking backgrounds. Adding on, there may be regions in the background with skin (Fig. 2.4k). In such a scenario, selection of salient fingers becomes a tedious task.

- **Blur:** During the capture process, a common problem is unfocused acquisition of an image. It may lead to a blurred finger-selfie due to which ridge-valley details might not be prominent. Similarly, finger-selfie may incur motion blur due to hand movement or unstable holding of smartphones.
- **Deformation:** In some cases, participants provided finger-selfies with crooked fingers.

2.3.4 Ground-Truth Annotation

Due to various challenges incorporated in the proposed database (as mentioned in Sect. 2.3.3), the position and appearance of fingers in the images vary. To determine the exact location of the finger, it is necessary to generate ground-truth annotations for the same. A segmentation tool is developed in MATLAB using Piotr Dollar's toolbox [30]. The GUI of the toolbox allows the user to utilize rotatable and resizable rectangular boxes to manually bound the finger region. With a rectangular region representing a finger region, only a minimal amount of background pixels are labeled as foreground. It acts as a loose bound for the finger, making sure that there is only a negligible loss of ridge-valley details. The rectangular region can easily be cropped and fed to recognition modules. The ground-truth annotations, which are represented as a mask, are also publicly available along with the database with the same image name in a different folder.

2.3.5 Experimental Protocol

As mentioned in Sect. 2.3.2, the UNFIT database is collected from 115 subjects with 30 images taken from each participant. While training and testing, a 50:50 subject disjoint split is maintained. Hence, training data includes 1740 images corresponding to 58 subjects, and testing data consists of remaining 1710 images from 57 participants. The index and middle fingers of the same subject are considered as different classes, resulting in 116 classes during training and 114 classes while testing. During testing, the first five images of each case (index, middle, or both fingers) are treated as the gallery, whereas the remaining images (sample #6–10) are considered as the query images. While generating scores, the genuine scores are generated when index–index, middle–middle, and multiple–multiple fingers of the same subjects are matched. All other combinations of match scores generated by matching query with gallery images are treated as imposter scores.

2.4 Segmentation Framework

The unique and discriminative features of a fingerprint lie in its ridge-valley pattern. These details are present on the finger-tip, which constitutes for the foreground of the finger-selfie image. Hence, a framework is presented which aims to discard the background pixels and keep only the foreground information. A summary of the segmentation framework is illustrated in Fig. 2.7, and its details are elaborated below.

2.4.1 Segmentation Using VGG SegNet

The segmentation framework primarily utilizes VGG SegNet for classifying pixels as foreground or background. The VGG SegNet architecture has encoder and decoder network. While the role of the encoder is to convert the input data into a meaningful feature map at a lower dimension, the decoder upsamples the lower-dimensional feature map. The lower-dimensional feature map is produced due to max-pooling operation after a sequential process of convolution, batch normalization, and ReLU activation to produce nonlinearity. The locations of features, which are propagated in the network after max-pooling, are stored for further computation.

The decoder network utilizes pooling indices (the ones stored during encoding) to perform a nonlinear upsampling in order to counter the effect of max-pooling. The stored pooling indices guide the decoder network to map a lower input feature map to a higher-dimensional feature map. Hence, the upsampled feature map obtained from the decoder network has a sparse representation of the input. The upsampling approach using pooling indices is a training-free method, hence reducing the number of training parameters of the model.

While pooling is known to have local invariance, in this work, a standard encoder–decoder network with pooling layers is utilized. The previous encoder–decoder architectures also use a standard pooling in their model (or global average pooling at the end of the network). It can be noted that networks that have used pooling [11, 31, 32]

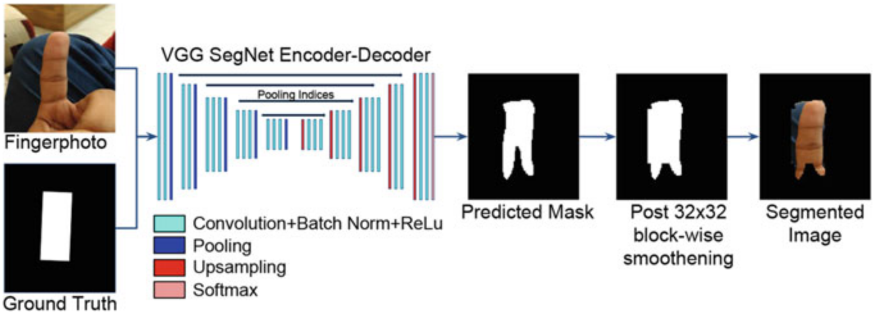


Fig. 2.7 Illustration of the segmentation framework using VGG SegNet followed by 32 × 32 block-wise smoothening

have worked well for the task of object segmentation. However, to eliminate pooling, the entire model has to be revamped and replaced by a capsule-net style architecture. Such scenario would require training from scratch, disallowing us to use pre-trained network. With a limited number of training instances, training a pooling-free network would be beyond the scope of the proposed framework.

The sparse representation is fed as input to a convolutional layer, which is succeeded by a Softmax classification layer. The Softmax layer classifies each of the image pixels as foreground or background. Thus, the VGG SegNet-based segmentation algorithm utilizes a pre-trained model of VGG SegNet for finger-selfie segmentation. The model is fine-tuned using finger-selfies. However, as we explain in Sect. 2.4.4.1, the predicted mask is tightly bound, due to which a significant foreground area is lost. Therefore, VGG SegNet architecture is succeeded by a 32 × 32 block-wise smoothening layer to increase the number of foreground pixels. The full segmentation pipeline is shown in Fig. 2.7. Algorithm 1 summarizes the complete segmentation algorithm.

```
  Input: 224×224 finger-selfie image
  Output: Segmented mask for finger-selfie
Fine-tune VGG SegNet Architecture using training finger-selfies and their masks;
Use trained model to predict mask for test finger-selfies;
Binarize the predicted masks;
$f_p$ = Count of finger (foreground) pixels;
$b_p$ = Count of non-finger (background) pixels;
N = Number of test images;
Region = Number of non-overlapping blocks of dimension 32×32 pixels in a finger-selfie;
while $N \neq 0$ do
    Divide test image into blocks of size 32×32 pixels;
    while Region do
        if $f_p \geq b_p$ then
            Set all pixels of the region as foreground;
        else
            Do not update any pixels of the region;
        end
        Region = Region - 1;
    end
    $N = N - 1$;
end
```

Algorithm 1: Algorithm for finger-selfie segmentation using a fine-tuned VGG SegNet architecture followed by a layer of 32 × 32 block-wise smoothening.

2.4.2 Implementation Details

To train the VGG SegNet + 32 × 32 block-wise smoothening network, finger-selfies of size 224 × 224 × 3 are used along with their corresponding ground-truth annotation of size 224 × 224 × 1. As illustrated in Fig. 2.7, VGG SegNet consists of an encoder and a decoder network. The output dimension of encoder network is

14 × 14 × 512. This multi-channel output is fed to the decoder network, which in turn gives an output of dimension 112 × 112 × 2. The output of the decoder network serves as input to the Softmax layer, whose task is to provide a binary prediction for each pixel. The white pixel in the binary predicted mask represents the finger region, whereas the black pixel represents the background. Similar to VGG SegNet, FCN 8 is also provided finger-selfies and its corresponding ground-truth annotation.

The VGG SegNet and FCN 8 architectures are fine-tuned using an augmented training set. The augmented training data is created by increasing the original training set with mirror flipped, intensity changed, blurred, and rotated finger-selfies. Rotation of finger-selfies is performed at three different angles: 90°, 180°, and 270°. After image augmentation, the size of the training set increases to 27600 images. The corresponding finger location annotation is generated for these augmented images from the original ground-truth annotation. Using the augmented training dataset, the deep architectures are fine-tuned for 100 epochs.

2.4.3 Performance Evaluation Metrics

To evaluate the performance of segmentation algorithm, the following metrics are used:

- Segmentation accuracy (SA):

$$SA = \frac{CPB}{TB} \tag{2.1}$$

  where CPB is a count of the correctly predicted blocks while TB is the total number of blocks.
- Foreground segmentation accuracy (FSA):

$$FSA = \frac{CPFB}{TFB} \tag{2.2}$$

  FSA is the normalized foreground segmentation accuracy, where CPFB represents the number of correctly predicted foreground blocks, normalized with respect to the total count of foreground annotated blocks (TFB).
- Background Segmentation Accuracy (BSA):

$$BSA = \frac{CPBB}{TBB} \tag{2.3}$$

  BSA is the normalized background segmentation accuracy, where CPBB portrays the number of correctly predicted background blocks normalized with respect to the total count of background annotated blocks (TBB).

Figure 2.8 demonstrates a visual elucidation of FSA and BSA using the segmentation algorithm.

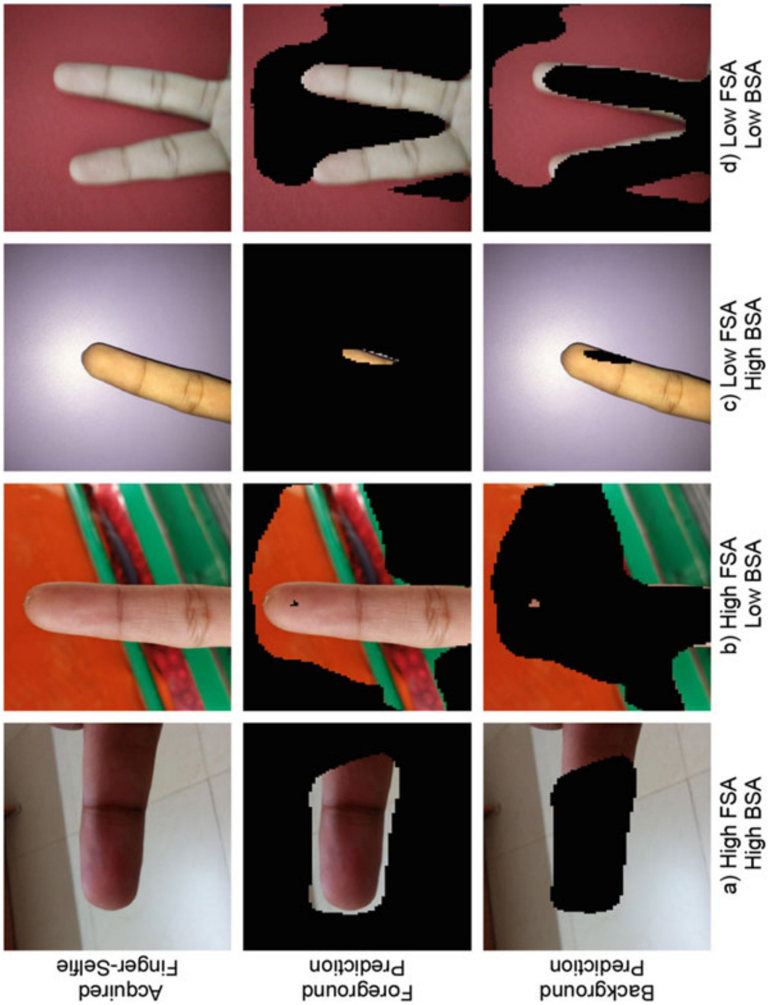


Fig. 2.8 Interpretation of FSA and BSA while segmenting finger-selfies

2.4.4 Segmentation Performance

Table 2.3 reports the segmentation performance of the algorithm in terms of FSA, BSA, and SA. VGG SegNet, along with 32 × 32 block-wise smoothening, provides the best foreground segmentation accuracy and performs well in terms of BSA and SA as well. Tables 2.4 and 2.5 illustrate a comparison of various segmentation techniques with the VGG SegNet+block-wise smoothening algorithm. Figure 2.9 shows a few samples where the segmentation framework can segment finger-selfie correctly, whereas Fig. 2.10 shows some failure cases of the segmentation algorithm.

In the proposed UNFIT database, background pixels constitute 86.21% pixels compared to 13.79% foreground pixels. While FSA is lower than BSA in Table 2.3, the reported segmentation accuracy (SA) is biased toward BSA for all fingers. This is due to higher number of background pixels in the UNFIT database as compared to foreground finger region pixels.

2.4.4.1 Effect of 32 × 32 Block-Wise Smoothening

Table 2.4 shows a comparison of the proposed architecture with VGG SegNet. For VGG SegNet, it can be observed that BSA outperforms FSA for all the fingers. The

Table 2.3 Segmentation performance of the VGG SegNet + 32 × 32 block-wise smoothening finger-selfie segmentation algorithm

Algorithm	Segmentation metric	Finger			
		All together (%)	Index (%)	Middle (%)	Multiple (%)
VGG SegNet + 32 × 32 block-wise smoothening	SA	89.04	89.89	90.62	86.61
	BSA	92.71	93.16	93.06	91.91
	FSA	71.22	70.28	74.49	68.90

Table 2.4 Comparison of the segmentation framework with VGG SegNet: illustrating the effect of 32 × 32 block-wise smoothening

Algorithm	Segmentation Metric	Finger			
		All together (%)	Index (%)	Middle (%)	Multiple (%)
VGG SegNet	SA	90.08	91.01	91.77	87.45
	BSA	94.69	95.04	94.89	94.15
	FSA	66.75	65.98	70.16	64.10
VGG SegNet + 32 × 32 block-wise smoothening	SA	89.04	89.89	90.62	86.61
	BSA	92.71	93.16	93.06	91.91
	FSA	**71.22**	**70.28**	**74.49**	**68.90**

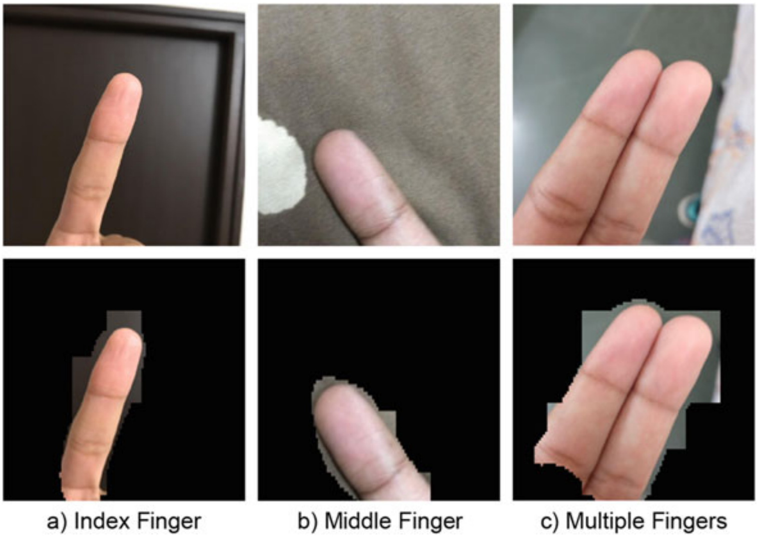


Fig. 2.9 Illustration of the successful cases of the segmentation framework

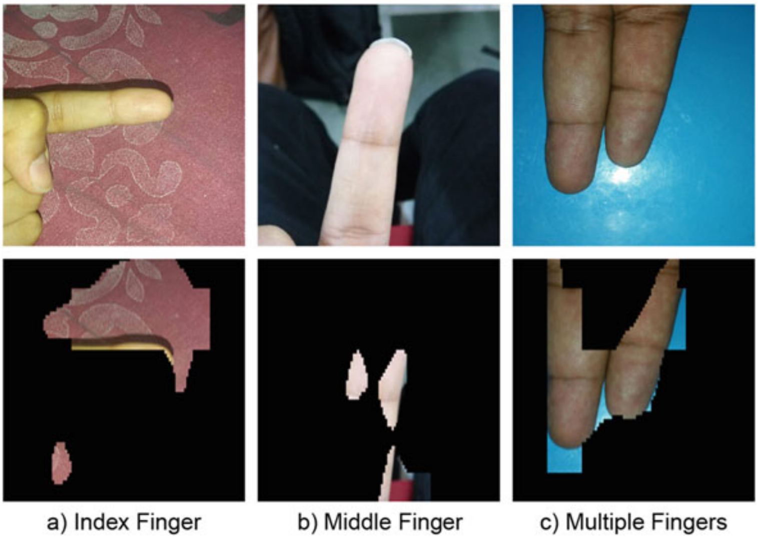


Fig. 2.10 Illustration of the failure cases of the segmentation framework

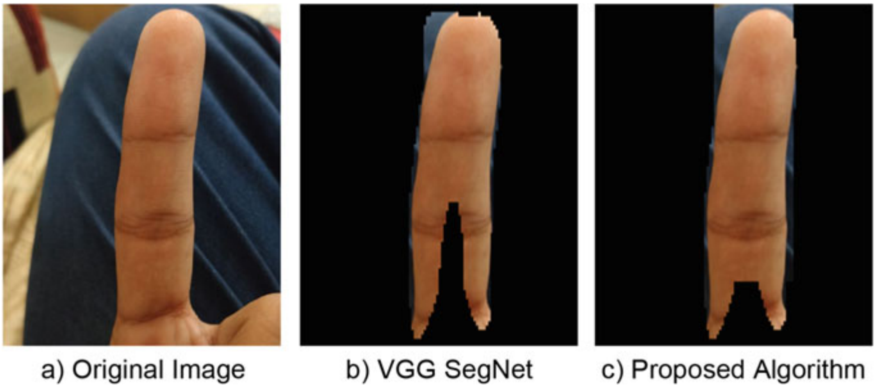


Fig. 2.11 Significance of 32 × 32 smoothening over VGG SegNet architecture

reason for higher BSA is the tight bound over the located finger-selfie obtained by the trained VGG SegNet. A drawback of a tight bound over the located finger-selfie is that few foreground finger regions are termed as background while most background regions are predicted as background. Thus, for VGG SegNet, BSA is higher than FSA due to erroneous classification of foreground pixels on the boundary of the located finger-selfie.

As observed by the segmentation performance of VGG SegNet in Table 2.4, FSA remains lower due to misclassification of foreground pixels located on the boundary of the located finger-selfie. Loosening the predicted boundary by VGG SegNet will increase foreground pixels, in turn increasing FSA. Thus, a 32 × 32 block-wise smoothening layer is added in the VGG SegNet architecture and it aids in increasing the FSA from 66.75 to 71.22%. While there is a trade-off for reduced SA and BSA by 1.04 and 1.98%, respectively, the distinctive ridge-valley details present in foreground region in finger-selfies are not compromised. An illustration of the effect of smoothening over VGG SegNet is shown in Fig. 2.11.

2.4.4.2 Comparison of VGG SegNet with FCN 8

Similar to VGG SegNet, a FCN 8 architecture is also fine-tuned. Inferring from the positive effect of 32 × 32 block-wise smoothening on FSA, FCN 8 architecture also includes a 32 × 32 block-wise smoothening. The FCN 8 trains a fully convolutional encoder–decoder network, and it uses an AdaDelta optimizer and a cross-entropy loss function.

Table 2.5 shows a comparison of segmentation performance of FCN-8-based segmentation with VGG SegNet-based segmentation algorithm. However, with highest FSA and overall segmentation accuracy, the VGG SegNet + block-wise smoothening model outperforms under both the scenarios. One of the major reasons for better performance of VGG SegNet-based approach is the lesser number of trainable

Table 2.5 Comparison of segmentation performance of the finger-selfie segmentation framework with FCN 8

Algorithm	Segmentation metric	Finger			
		All together (%)	Index (%)	Middle (%)	Multiple (%)
FCN 8	SA	88.55	89.45	90.19	86.01
	BSA	93.92	94.22	94.09	93.45
	FSA	61.46	60.11	63.66	60.62
FCN 8 + 32 × 32 block-wise smoothening	SA	87.56	88.37	89.16	85.16
	BSA	92.04	92.41	92.43	91.27
	FSA	65.81	64.19	67.97	65.26
VGG SegNet + 32 × 32 block-wise smoothening	SA	89.04	89.89	90.62	86.61
	BSA	92.71	93.16	93.06	91.91
	FSA	71.22	70.28	74.49	68.90

parameters [33]. Using the max-pooling indices from respective encoding layers, the decoder in VGG SegNet performs sparse upsampling. This procedure reduces computation time as well as increases generalizability of the model. On the contrary, FCN 8 learns parameters for upsampling too. Hence, despite data augmentation, the training data may not be enough to train additional parameters, which justifies VGG SegNet outperforming FCN 8.

2.4.4.3 Comparison with Skin Color-Based Segmentation

Inspired from existing studies [12, 16, 18, 23], the VGG SegNet + 32 × 32 block-wise smoothening model is also compared with various skin color-based segmentation algorithms. The results are presented in Fig. 2.12. The foremost comparison is performed with a thresholding color channel-based skin color segmentation algorithm [34, 35]. The finger-selfie image, available in RGB color space, is converted to HSV and YCbCr color space. The information in Hue, Cb, and Cr color space is used to find probable skin color regions using pre-defined thresholds. While the VGG SegNet + 32×32 block-wise smoothening method provides FSA of 71.22%, skin color-based segmentation provides FSA of 58%. Segmentation algorithm proposed by Sankaran et al. [12] also fails to perform well. Due to image augmentation by varying intensities, our fine-tuned model becomes robust toward illumination variations and flash usage in finger-selfies. However, because of too bright or too dull skin regions in certain cases, the standard skin color algorithms fail due to fixed thresholds.

Additionally, a comparison is shown of skin color segmentation with a deep architecture. Firstly, the salient region is cropped out using skin color-based segmentation. The salient region is fed as input to the architecture: VGG SegNet + 32 × 32

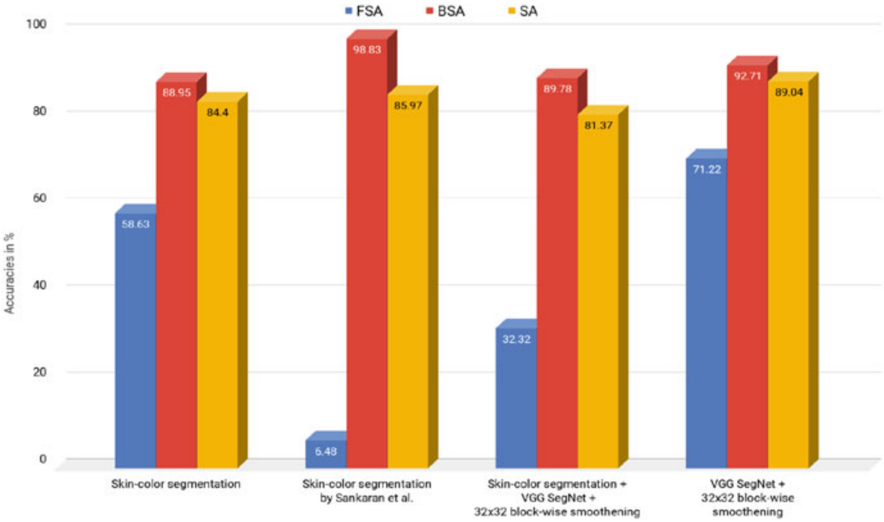


Fig. 2.12 Comparison of segmentation accuracies obtained with the skin color-based techniques and the VGG SegNet with block-wise smoothening algorithm

smoothening. However, both SA and FSA are reduced. The results are shown in Fig. 2.12. These results indicate that in an unconstrained scenario, skin color-based segmentation is likely to fail.

2.5 Finger-Selfie Recognition

In 2013, Li et al. [22] highlighted that minutiae-based techniques for feature extraction and matching would fail for finger-selfies. Sankaran et al. [12] showcased a similar inference, highlighting that minutiae-based techniques fail for semi-constrained scenarios. Hence, the authors used ScatNet for their experiments. While ScatNet worked for the semi-constrained scenario, the representation would fail to encode discriminatory information under deformations and rotational variations present in the UNFIT database. As a result, we too utilized two non-minutiae-based algorithms for feature extraction, namely CompCode and ResNet50. The details are mentioned in the subsection below.

2.5.1 Feature Representations

Non-Deep learning: Competitive Coding (CompCode) [13, 36] is a popular non-minutiae-based feature representation, commonly deployed for fingerprint and palmprint recognition. Quite recently, CompCode and its variant were exploited for utilizing ridge-valley details present in palmprints for person recognition [36]. With ridge-valley pattern forming a unique structure, filters that encode orientation information can provide an efficient feature representation. CompCode features are extracted by convolution of the real part of the Gabor filter G_r over the image I. The Gabor filters G_r have J different orientations, each of which varies from previous by $\frac{\pi}{J}$. Along with orientation variations, Gabor filters differ in frequency W as well. Hence, the total number of filters convolved to obtain the feature representation are $J \times W$. The response of the filter, convolved over the segmented finger-selfie I, is given as:

$$R = I(x, y) * \psi_R(x, y, \omega_i, \theta_j) \tag{2.4}$$

Here, ψ_R is the real part of the Gabor filter ψ, while ω_i and θ_j are frequency and orientation of the Gabor filter. Note that the segmented output is upscaled to a fixed size of 400×400 before applying Gabor filters to obtain the representation.

Deep learning-based approach: The segmented finger-selfie is served as input to the ResNet50 architecture [14]. ResNets have shown their application to general object recognition with deeper networks. To counter the effect of vanishing gradient and overfitting, ResNets have shortcut connections among different convolutional layers. Intuitively, along with the feedforward mapping $F(x)$ from the previous layer C_l, the input to the next convolution layer C_{l+1} also includes an identity mapping x from some previous layer C_{l-k}. Hence, the input to convolutional layer C_{l+1} can be written as:

$$F(x, \{W_i\}) + x \tag{2.5}$$

where W_i signifies transformation through multiple convolutional layers. In the ResNet50 architecture, the function $F(x)$ involves two stacked convolutional layers. This implies that the input x is taken from the activated output of layer C_{l-2}, and $W_i x$ is a transformation of x over two convolutional layers.

The segmented RGB image is provided to the network at a fixed size of 224 × 224. In our experiments, the ResNet50 architecture is initialized using the weights of the model trained on the ImageNet database. With the Softmax classification layer removed, the network provides a feature vector of dimension 2048 × 1, which is treated as the feature representation of the finger-selfie. The intermediate layers of

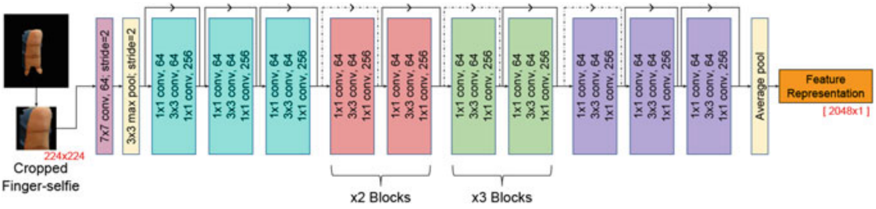

224x224
Cropped Finger-selfie
7x7 conv, 64; stride=2
3x3 max pool; stride=2
1x1 conv, 64 3x3 conv, 64 1x1 conv, 256
1x1 conv, 64 3x3 conv, 64 1x1 conv, 256
1x1 conv, 64 3x3 conv, 64 1x1 conv, 256
1x1 conv, 64 3x3 conv, 64 1x1 conv, 256
1x1 conv, 64 3x3 conv, 64 1x1 conv, 256
1x1 conv, 64 3x3 conv, 64 1x1 conv, 256
1x1 conv, 64 3x3 conv, 64 1x1 conv, 256
1x1 conv, 64 3x3 conv, 64 1x1 conv, 256
1x1 conv, 64 3x3 conv, 64 1x1 conv, 256
1x1 conv, 64 3x3 conv, 64 1x1 conv, 256
x2 Blocks
x3 Blocks
Average pool
Feature Representation
[2048x1]

Fig. 2.13 Procedure to obtain feature representation using ResNet50 architecture

ResNet50 look for different shapes and strokes. Hence, the final feature representation encodes curves, vertical and horizontal lines, and other shapes, which are equivalent to ridge orientations, finger shape, and phalanx lines. The procedure to obtain the feature representation is illustrated in Fig. 2.13.

2.5.2 Finger-Selfie Recognition Performance

After extracting features from finger-selfie images, the next step is to match the query feature templates with the gallery templates. The CompCode features are matched with gallery templates using Hamming distance to obtain a distance score. Similarly, representation obtained from ResNet50 architecture is matched with gallery templates using cosine similarity.

On the testing set of 57 subjects, receiver operating characteristic (ROC) curve is used to report the verification performance. The ROC curve is shown in Fig. 2.14. Table 2.6 shows the confusion matrix when feature representation from CompCode and ResNet50 are matched using Hamming distance and cosine similarity, respectively. In spite of the potency of CompCode for palmprint and fingerprint recognition, we observe an EER of 41.41% for finger-selfie matching. On the other hand, the cosine similarity of ResNet50-based representation yields a better performance with EER as 35.32%.

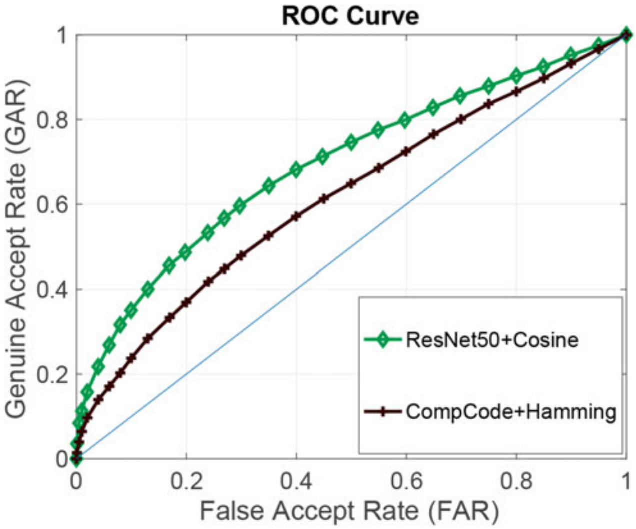


Fig. 2.14 Receiver operating characteristic (ROC) curve for the VGG SegNet + 32 × 32 segmentation pipeline. Representation from ResNet50 architecture is matched using cosine similarity, and CompCode features are matched using Hamming distance metric on the test set of UNFIT database

Table 2.6 Confusion matrix when feature representation from CompCode and ResNet50 are matched using Hamming distance and cosine similarity, respectively. From a total of 731,025 pairs (855 probe representations matched with 855 gallery representations), there are 4275 genuine and 726,750 imposter pairs. Values are reported at 10% FAR

		Prediction			
		CompCode+Hamming		ResNet50+Cosine	
		Match	Non-Match	Match	Non-Match
Ground truth	Match	1016	3259	1517	2758
	Non-match	72,480	654,270	73,335	653,415

The finger-selfie dataset, namely the UNFIT database, has numerous variations. The variations occur due to an unconstrained environment. The ResNet50 model is pre-trained on ImageNet database, where objects are of different shapes and sizes. These learned weights can handle variations in finger-selfies pertaining to scale and orientation of finger-selfie. Also, ResNet50 feature representation for segmented finger-selfies is matched using cosine similarity. Since cosine similarity is an angular similarity of two vectors, variations introduced in the magnitude of representations due to illumination variations do not effect cosine similarity. Hence, the recognition model becomes robust toward illumination variations. Thus, the overall performance of ResNet50 + Cosine similarity is better than CompCode + Hamming distance-based recognition.

While such results are motivating that deep architectures have a better potential for finger-selfie recognition, there is still a long way to go for recognition of finger-selfies in an unconstrained scenario. With the proposed UNFIT database, we expect that the research community will be driven toward building better segmentation, enhancement, quality assessment, and feature representation modules for finger-selfie-based recognition.

2.6 Conclusion

This chapter presents a review of existing research on finger-selfies and later introduces finger-selfie in an unconstrained environment. The proposed UNconstrained FIngerphoTo (UNFIT) database incorporates various challenges such as rotation, translation, orientation, position, scale, multiple fingers, illumination, background, and resolution which arise due to the differing environments in which the finger-selfies are acquired. This database includes the manual annotations and experimental protocol, using which segmentation and verification results are benchmarked. A VGG SegNet-based segmentation approach is presented along with baseline results, followed by matching algorithms using CompCode and ResNet50 representations. We assert that the proposed database can take forward the research in this domain and the segmentation pipeline can segment and perform authentication using finger-selfies despite the challenges posed in the database. Future work can include quality assessment for detection of poor-quality finger-selfies and use of minutiae in conjunction with deep learning features for improved recognition performance.

Acknowledgements The authors would like to thank all the volunteers for their enthusiastic participation in the collection of the UNFIT database. Aakarsh Malhotra is partly supported through the Visvesvaraya Ph.D. Scheme of Ministry of Electronics & Information Technology, Government of India, being implemented by Digital India Corporation. Mayank Vatsa and Richa Singh are partly supported by the Infosys Center for Artificial Intelligence, IIIT Delhi, India. Mayank Vatsa is also supported by Swarnajayanti Fellowship from Government of India.

References

1. Taekyoung K, Jin H (2015) Analysis and improvement of a PIN-entry method resilient to shoulder-surfing and recording attacks. IEEE Trans Inf Forensics Secur 10(2):278–292
2. Staff M, Fleishman G (2018) iPhone X. https://www.macworld.com/article/3225406/iphone-ipad/face-id-iphone-x-faq.html. Accessed on 11 Feb 2018
3. Bajaj K, Bhagat HR (2018) Your phone's fingerprint scanner can do much more than just unlock your phone. Here's how . https://economictimes.indiatimes.com/magazines/panache/your-phones-fingerprint-scanner-can-do-much-more-than-just-unlock-your-phone-heres-how/articleshow/57766012.cms. Accessed on 02 May 2018
4. Tim Ahonen. Phone book 2012: Statistics and facts on the mobile phone industry, 2012. http://www.tomiahonen.com/ebook/phonebook.html. Accessed on 02 May 2018
5. Richter F (2018) Smartphones cause photography boom. https://www.statista.com/chart/10913/number-of-photos-taken-worldwide/. Accessed on 20 June 2018
6. Kumar A, Zhou Y (2011) Contactless fingerprint identification using level zero features. In: IEEE conference on computer vision and pattern recognition workshops, pp 114–119
7. Wood C (2018) WhatsApp photo drug dealer caught by 'groundbreaking' work. http://www.bbc.com/news/uk-wales-43711477. Accessed on 25 May 2018
8. Hern A (2018) Hacker fakes German minister's fingerprints using photos of her hands. https://www.theguardian.com/technology/2014/dec/30/hacker-fakes-german-ministers-fingerprints-using-photos-of-her-hands. Accessed on 25 May 2018
9. Chopra S, Malhotra A, Vatsa M, Singh R (2018) Unconstrained fingerphoto database. In: IEEE conference on computer vision and pattern recognition workshops, pp 517–525
10. Badrinarayanan V, Kendall A, Cipolla R (2015) SegNet: a deep convolutional encoder-decoder architecture for image segmentation. In arXiv:1511.00561v3
11. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: IEEE conference computer vis pattern tecognit, pp 3431–3440
12. Sankaran A, Malhotra A, Mittal A, Vatsa M, Singh R (2015) On smartphone camera based fingerphoto authentication. In: IEEE international conference on biometrics theory, applications and systems, pp 1–7
13. Kong AW-K, Zhang D (2004) Competitive coding scheme for palmprint verification. In: IAPR international conference on pattern recognition vol 1, pp 520–523
14. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: IEEE conference on computer vision and pattern recognition, pp 770–778
15. Song Y, Lee C, Kim J (2004) A new scheme for touchless fingerprint recognition system. In: IEEE international symposium on intelligent signal processing and communication systems, pp 524–527
16. Lee C, Lee S, Kim J, Kim S-J (2006) Preprocessing of a fingerprint image captured with a mobile camera. In: IAPR international conference on biometrics. Springer, pp 348–355

17. Lee D, Choi K, Choi H, Kim J (2008) Recognizable-image selection for fingerprint recognition with a mobile-device camera. IEEE Trans Syst, Man, Cybern, Part B (Cybern) 38(1):233–243
18. Piuri V, Scotti F (2008) Fingerprint biometrics via low-cost sensors and webcams. In: IEEE international conference on biometrics: theory, applications and systems, pp 1–6
19. Hiew BY, Teoh ABJ, Yin OS (2010) A secure digital camera based fingerprint verification system. J Vis Commun Image Represent 21(3):219–231
20. Derawi MO, Yang B, Busch C (2011) Fingerprint recognition with embedded cameras on mobile phones. In: International conference on security and privacy in mobile information and communication systems. Springer, pp 136–147
21. Yang B, Li G, Busch C (2013) Qualifying fingerprint samples captured by smartphone cameras. In: IEEE international conference on image processing, pp 4161–4165
22. Li G, Yang B, Olsen MA, Busch C (2013) Quality assessment for fingerprints collected by smartphone cameras. In: IEEE conference on computer vision and pattern recognition workshops, pp 146–153
23. Raghavendra R, Busch C, Yang B (2013) Scaling-robust fingerprint verification with smartphone camera in real-life scenarios. In: IEEE international conference on biometrics: theory, applications and systems, pp 1–8
24. Stein C, Bouatou V, Busch C (2013) Video-based fingerphoto recognition with anti-spoofing techniques with smartphone cameras. In: IEEE international conference of the biometrics special interest group, pp 1–12
25. Tiwari K, Gupta P (2015) A touch-less fingerphoto recognition system for mobile hand-held devices. In: IAPR international conference on biometrics, pp 151–156
26. Taneja A, Tayal A, Malhotra A, Sankaran A, Vatsa M, Singh R (2016) Fingerphoto spoofing in mobile devices: a preliminary study. In: IEEE international conference on biometrics theory, applications and systems pp 1–7
27. Lin C, Kumar A (2018) Matching contactless and contact-based conventional fingerprint images for biometrics identification. IEEE Trans Image Process 27(4):2008–2021
28. Malhotra A, Sankaran A, Vatsa M, Singh R (2018) Learning representations for unconstrained fingerprint recognition. Deep Learn Biom 197–226
29. Malhotra A, Sankaran A, Mittal A, Vatsa M, Singh R (2017) Fingerphoto authentication using smartphone camera captured under varying environmental conditions. Human Recognition in Unconstrained Environments: Using Computer Vision, Pattern Recognition and Machine Learning Methods for Biometrics, pp 119–144
30. Dollár P (2018) Piotr's computer vision matlab tooflbox (PMT). https://github.com/pdollar/toolbox. Accessed on 22 Feb 2018
31. Girshick R (2015) Fast R-CNN. In: IEEE international conference on computer vision, pp 1440–1448
32. He K, Gkioxari G, Dollár P, Girshick R (2017) Mask R-CNN. In: IEEE international conference on computer vision, pp 2980–2988
33. Semantic Segmentation Models for Autonomous Vehicles. https://blog.playment.io/semantic-segmentation-models-autonomous-vehicles/. Accessed on 26 June 2018
34. Kakumanu P, Makrogiannis S, Bourbakis N (2007) A survey of skin-color modeling and detection methods. Pattern Recognit 40(3):1106-1122
35. Kolkur S, Kalbande D, Shimpi P, Bapat CP, Jatakia J (2016) Human skin detection using RGB, HSV and YCbCr color models. In: International Conference on Communication and Signal Processing
36. Zheng Q, Kumar A, Pan G (2016) Suspecting less and doing better: new insights on palmprint identification for faster and more accurate matching. IEEE Trans Inf Forensics Secur 11(3):633–641

Chapter 3
A Scheme for Fingerphoto Recognition in Smartphones

Ruggero Donida Labati, Angelo Genovese, Vincenzo Piuri and Fabio Scotti

Abstract Touchless technologies for fingerphoto recognition based on smartphones can be considered selfie biometrics, in which a user captures images of his or her own biometric traits by using the integrated camera in a mobile device (here referred to as selfie fingerprint biometrics). Such systems mitigate the limitations of leaving latent fingerprints, dirt on the acquisition device released by the fingers, and skin deformations induced by touching an acquisition surface associated with a touch ID-based system. Furthermore, the use of the integrated camera to perform biometric acquisition bypasses the need of a dedicated fingerprint scanner. With respect to touch-based fingerprint recognition systems, selfie fingerprint biometrics require ad hoc methods for most steps of the recognition process. This is because the images captured using smartphone cameras present more complex backgrounds, lower visibility of the ridges, reflections, perspective distortions, and nonuniform resolutions. Selfie fingerprint biometric methods are usually less accurate than touch-based methods, but their performance can be satisfactory for a wide variety of security applications. This chapter presents a comprehensive literature review of selfie fingerprint biometrics. First, we introduce selfie fingerprint biometrics and touchless fingerprint recognition methods. Second, we describe the technological aspects of the different steps of the recognition process. Third, we analyze and compare the performances of recent methods proposed in the literature.

R. Donida Labati (✉) · A. Genovese · V. Piuri · F. Scotti
Department of Computer Science, Università degli Studi di Milano, Via Celoria 18, 20133
Milano, Italy
e-mail: ruggero.donida@unimi.it

A. Genovese
e-mail: angelo.genovese@unimi.it

V. Piuri
e-mail: vincenzo.piuri@unimi.it

F. Scotti
e-mail: fabio.scotti@unimi.it

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision
and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_3

3.1 Introduction

Smartphones are a type of handheld mobile computer and are widely used all over the world to store sensitive data and to access a wide range of distributed services. Therefore, these devices require strong authentication mechanisms for properly managing access to local data and distributed information. For this reason, some recent smartphones possess integrated sets of sensors specifically designed for biometric acquisition, such as fingerprint scanners [17]; illumination systems, optics and cameras for ocular biometrics [39]; and devices for acquiring three-dimensional face samples [1]. In this context, fingerprint-based systems are particularly promising due to their high accuracy and their acceptance by users. However, not all current smartphones include a dedicated fingerprint scanner, whereas almost every smartphone includes a digital camera. Due to recent advances in the speed, resolution, and dynamic range of the digital cameras embedded in smartphones, selfie fingerprint biometrics is attracting increasing interest [3].

Touchless fingerphoto recognition technologies possess important advantages with respect to systems based on traditional touch-based fingerprint scanners: (i) the absence of elastic skin deformations, since the finger is not pressed onto any surface; (ii) the absence of latent fingerprints left on the sensor; (iii) the absence of dirt on the acquisition surface introduced by the touch-based acquisition process; and (iv) a faster capture of biometric data [16]. However, as shown in Fig. 3.1, the touchless fingerprint images acquired using the cameras integrated in smartphones can present several nonidealities in comparison to samples acquired using touch-based fingerprint scanners, as follows:

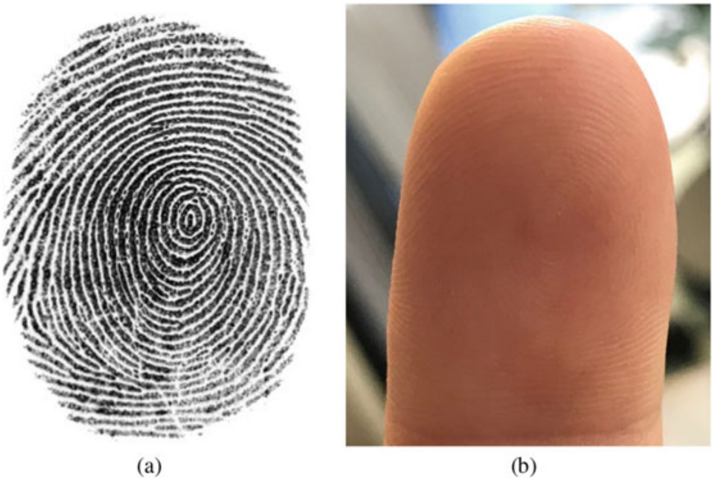


Fig. 3.1 Examples of fingerprint images of the same finger acquired using a touch-based sensor (**a**) and a smartphone camera (**b**). The touchless acquisition performed with a smartphone camera presents a more complex background, nonuniform illumination, and out-of-focus regions

- Such an image has a more complex background, including the skin of the finger, instead of containing only the ridge pattern as classical contact-based samples do;
- The illumination is not constant in all regions of the finger in the image;
- The fingerprint image contains reflections, reducing the contrast of the ridge pattern;
- The sample resolution varies because the distance from the finger to the camera is not constant among different acquisitions; thus, the application of the most commonly used fingerprint recognition algorithms (designed for samples with a standard resolution of 500 pixels per inch) becomes difficult;
- In many cases, the fingerprint image presents perspective distortions caused by uncontrolled rotations of the finger in three-dimensional space because no pins or references for finger positioning are provided during the acquisition process;
- The ridge pattern may not be sufficiently distinguishable in all regions of the fingerprint due to the limited depth of focus of the optics; and
- The sample may present motion blur.

Due to the aforementioned nonidealities, the accuracy of selfie fingerprint biometrics is currently inferior to that of traditional touch-based technologies [11, 16]. Selfie fingerprint biometric techniques require specifically designed algorithms for most steps of the recognition process.

This chapter is organized as follows. Section 3.2 describes the biometric recognition process of selfie fingerprint biometrics from a technological point of view, focusing on each step of the computational chain individually. Section 3.3 presents a performance analysis of state-of-the-art technologies. Finally, Sect. 3.4 concludes the work.

3.2 Biometric Recognition Process

Several studies in the literature have proposed touchless fingerprint acquisition systems based on a single camera [20, 35], multiple cameras [12, 24, 30, 46], or mobile devices (e.g., smartphone cameras) [4, 5, 42]. Recognition algorithms for touchless fingerprint samples can consider either two-dimensional images or three-dimensional models. While methods based on three-dimensional models can achieve a higher recognition accuracy than methods based on two-dimensional images can, they usually require complex acquisition setups, which are difficult to integrate into smartphones [11].

Selfie fingerprint biometric methods typically consider a single two-dimensional image, which can present the same nonidealities exhibited by touchless samples acquired using any digital camera. For this reason, many existing algorithms can be successfully applied to fingerprint images acquired using smartphone cameras as well as to samples acquired using other touchless devices. In addition, with respect to touchless systems based on a single camera or multiple cameras, mobile devices represent a more compact solution since all hardware and software components necessary to perform a correct acquisition are integrated in a single piece of equipment (e.g., a camera, focus assessment and correction functionalities, an illumination source, and a processing architecture).

The recognition procedure in selfie fingerprint biometrics usually consists of four steps: (i) acquisition, (ii) segmentation, (iii) enhancement, and (iv) feature extraction and matching. In addition, the recognition procedure can also include a quality assessment, a liveness detection, and a step for mitigating the nonidealities of touchless fingerprint sensors. Figure 3.2 shows a schema of the recognition process in selfie fingerprint biometrics.

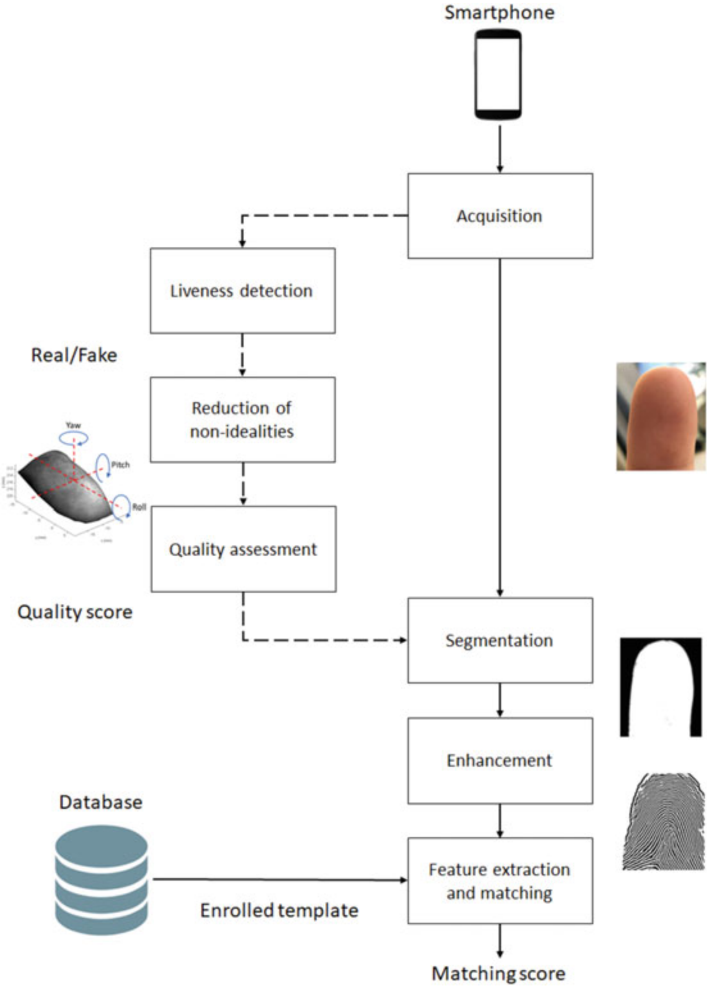


Fig. 3.2 Outline of the recognition process for selfie fingerprint biometrics

3.2.1 Acquisition

During the acquisition process, the biometric trait of interest is presented to the acquisition sensor, and a biometric sample is obtained. In the case of selfie fingerprint biometrics, one or more fingers are presented to the integrated camera of a smartphone, and the collected sample is a two-dimensional image. The acquisition methods presented in the literature feature important differences in terms of the techniques applied to control the finger positioning, illumination, and background. Fingerprint images acquired under controlled and uncontrolled conditions present important differences in terms of quality. In particular, acquisition procedures in which the finger positioning, illumination, and background are controlled can achieve better-quality images than uncontrolled acquisition techniques. However, controlled acquisition setups require a higher level of cooperation from the user. As an example, Fig. 3.3 shows fingerprint images of the same finger captured using a smartphone camera under controlled and uncontrolled conditions.

Several acquisition methods based on smartphone cameras are available. In the majority of cases, the camera parameters (focal distance, aperture of the diaphragm, and exposure time) are computed automatically by the acquisition software provided by the operating system, and the operator captures the fingerprint image as soon as the fingertip is within the field of view of the camera. The existing acquisition methods can be classified according to the number of fingers considered and the acquisition constraints applied (in terms of

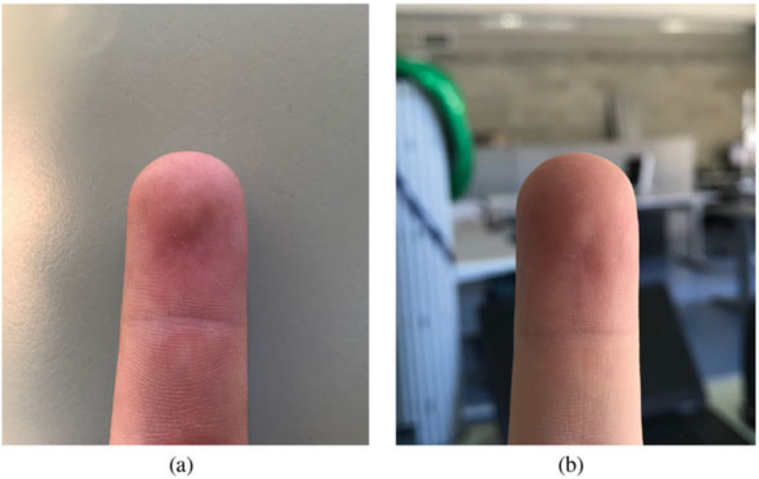


Fig. 3.3 Examples of fingerprint images captured using a smartphone camera under controlled (**a**) and uncontrolled (**b**) conditions. In images acquired under controlled conditions, the background is easier to remove, and the ridge pattern is more visible and less affected by noise. However, controlled acquisition setups are less usable than uncontrolled setups are and require a higher level of cooperation from the user

controlled or uncontrolled finger positioning and background). Specifically, it is possible to distinguish five classes: (i) single fingerprints with controlled finger positioning, controlled background, and illumination conditions; (ii) single fingerprints with uncontrolled finger positioning but controlled background and illumination conditions; (iii) single fingerprints with uncontrolled finger positioning, uncontrolled background, and illumination conditions; (iv) multiple fingerprints with controlled finger positioning but uncontrolled background and illumination conditions; and (v) multiple fingerprints with uncontrolled finger positioning, uncontrolled background, and illumination conditions.

In acquisition setups discussed in class (i), images are acquired under laboratory conditions; supports are used to position the smartphone, dedicated illumination setups are used, and the user is required to place his or her finger on a flat surface [6].

In acquisition methods discussed in (ii), constraints on the position of the finger are reduced but the background and illumination conditions are controlled [42, 44]. In these setups, the operator (who may also be the owner of the fingerprint) holds the device, while the software installed on the smartphone automatically captures the image. The LED of the smartphone is used as a flashlight to enhance the details of the fingerprint and the contrast with the background, making it easier to segment the region of interest in the image.

In acquisition setups discussed in (iii), the constraints are further reduced; consequently, such methods must cope with various uncontrolled backgrounds captured under both indoor and outdoor conditions [36]. There are publicly available datasets of fingerprint images captured under both indoor and outdoor conditions with controlled and uncontrolled backgrounds [21, 40].

While most smartphone-based acquisition procedures focus on a single finger at a time, the acquisition methods discussed in (iv) use multifinger acquisition setups that require previously defined procedures for positioning the fingers [4]. In such an acquisition procedure, a translucent guide is superimposed on the screen of the device to help the user both in correctly positioning the fingers and in capturing images with a constant distance between the fingers and the camera, thereby ensuring a fixed resolution.

The acquisition setups discussed in (v) need to overcome all possible nonidealities of the samples due to an unconstrained acquisition setup. There is a publicly available database of fingerprint samples acquired using smartphones consisting of images collected without any constraints on position, illumination, background, focus, or the number of fingers [22]. These fingerprint images present high variability since they were acquired using different cameras and acquisition software.

3.2.2 Segmentation

The purpose of the segmentation step is to separate the biometric trait of interest from other information in the sample. In the case of selfie fingerprint biometrics, this step aims to extract the region corresponding to the ridge pattern of the last phalanx. The proposed segmentation approaches in the literature can be divided into those based on samples acquired in controlled (i) or uncontrolled (ii) backgrounds. While in the first case, it is often possible to use general-purpose segmentation approaches, in the second case, it is necessary to consider additional challenges, such as the properties of the skin color or the presence of out-of-focus regions.

An example of a lightweight method for images acquired with controlled backgrounds is to threshold the red channel of the image to detect the region in which the finger is present [42]. Adaptive thresholding techniques can be applied to color as well as grayscale images [44]; for example, a background subtraction method can be used in combination with a thresholding technique based on the skin color [4].

Segmenting images with uncontrolled backgrounds require methods that are more complex than those based on controlled backgrounds. For example, an algorithmic segmentation approach that consists of a preliminary training step and subsequent refinement steps for background removal is described in [25]. The preliminary training step collects information related to the distribution of the skin pixels in the RGB color space. The first refinement step builds a look-up table using the color distribution information and performs a color-based segmentation to determine whether the pixel belongs to the finger region or not. The second refinement step exploits the frequency information and computes a second segmentation mask by assuming that the regions of the image that do not correspond to the finger are out of focus and therefore contain only limited information at low frequencies. The last step combines the color- and frequency-based results using a region growing algorithm.

Algorithms for skin color detection can also be applied to segment images with uncontrolled backgrounds. A well-known method for skin detection used for segmenting touchless fingerprint images acquired using smartphones is the mean shift segmentation. In this method, several segments, each corresponding to a different region of the image, are compared against a fixed reference image to correctly establish which region depicts the finger [23].

Skin detection algorithms may also rely on thresholding channels of the image in color spaces other than the most frequently used RGB color space. Fingerprint segmentation in images acquired using smartphones with uncontrolled backgrounds can be performed by thresholding the magenta (M) channel in the CMYK color space [40] or by thresholding a combination of channels in the YCbCr and HSV color spaces [2].

Recently, deep learning and convolutional neural networks (CNNs) are being increasingly used for a wide variety of signal and image processing applications, including the extraction of relevant information from biometric samples [7]. CNNs can also be successfully applied to segment touchless fingerprint images with uncontrolled backgrounds [5].

In the case of acquisitions with multiple fingers, it is possible either to separate the fingers such that they can be individually matched or to perform multimodal sample-level fusion [37] by treating each multifinger acquisition as a single biometric sample [5]. The separation of different fingers can be performed by estimating the boundaries between the fingers using edge detectors [4].

3.2.3 Enhancement

The enhancement step aims to reduce noise and improve the distinguishability of the distinctive characteristics of a biometric trait. In the case of selfie fingerprint biometrics, the enhancement step is performed in most of the systems and has the purposes of improving the visibility of the ridge pattern and removing unnecessary details in the image. There are two main classes of enhancement techniques applicable to touchless fingerprint images acquired using smartphones: (i) those that enhance the visibility of the ridges using reduced computational resources and (ii) schemes that aim to obtain an enhanced representation of the ridge pattern that is as similar as possible to touch-based samples. As an example, Fig. 3.4 shows a touchless fingerprint image captured with a smartphone camera and the corresponding enhanced representation with minutiae features [33] extracted using a commercial software designed for touch-based samples [34].

Schemes based on enhancing the visibility of the ridges use well-known image processing algorithms, such as Wiener filtering [36] and adaptive histogram equalization [44], which are applied to perform fast computations and enhance the visibility of the ridges.

Schemes aiming to obtain a representation of the ridge pattern similar to touch-based samples are more computationally expensive and usually incorporate two tasks: noise reduction and enhancement of the ridge pattern. The noise reduction task can be performed by applying a median filter [42], followed by histogram equalization [40], and a band-pass filter tuned to the frequency of the ridges [4], or a processing sequence consisting of a Wiener low-pass filter, a top-hat filter, and histogram

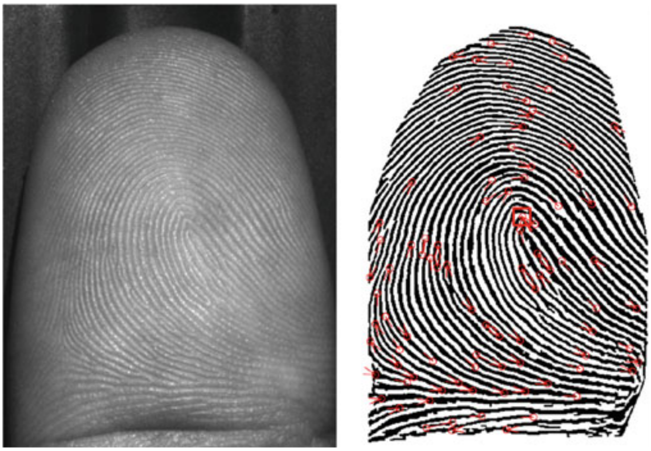


Fig. 3.4 Example of a touchless fingerprint image captured with a smartphone camera (**a**) and the corresponding enhanced representation (**b**). This figure shows that commercial software [34] can successfully extract the minutiae features [33] from an enhanced touchless fingerprint representation

equalization [31]. The visibility of the ridge pattern can be enhanced by means of an adaptive binarization procedure [42], an unsharp masking algorithm followed by local histogram normalization [40], or a set of Gabor filters tuned according to the local frequency and orientation of the ridges [31].

The proposed schemes may also perform the enhancement step as one single task. As an example, a bank of wavelets can be used to estimate the phase congruency of the frequency response and to extract the local regions of the image with higher phase congruency, which are then identified as parts of the ridge pattern [2].

3.2.4 Feature Extraction and Matching

The feature extraction step aims to extract a digital representation of unique features from a biometric sample (called a template), while the purpose of matching is to compute a similarity or dissimilarity score between two or more templates (called a matching score or a distance, respectively). In the case of selfie fingerprint biometrics, the methods in the literature can be classified according to the used feature sets: (i) Level 1 features, (ii) Level 2 features, and (iii) learned features. Level 1 features are global characteristics of the ridge pattern [33]. Level 2 features are local characteristics describing certain formations of the ridges, namely ridge endings or bifurcations, also called minutiae [33]. Learned features cannot be classified as pertaining to Level 1 or Level 2 since they are automatically learned from training data; they can be extracted using a variety of computational intelligence techniques, such as artificial neural networks, support vector machines, CNNs, deep neural networks and dictionary-based techniques [18].

Level 1 features are typically designed for touchless fingerprint samples but can also be applied to images acquired using smartphone cameras. There are methods for extracting feature vectors that describe the ridge orientation flow using Gabor filters [19] and methods for extracting singular points from touchless fingerprint samples [8].

Most of the methods in the literature adopt feature extraction and matching techniques pertaining to Level 2. Minutiae-based feature extractors and matchers can be directly applied to touchless fingerprint images acquired using smartphones [6, 27]. However, most of the methods in the literature extract minutiae-based features from enhanced ridge pattern images to achieve better accuracy. To extract and match minutiae-based features, commercial biometric recognition software tools designed for touch-based samples are widely used, with satisfactory results [2, 4, 42]. Furthermore, the minutiae-based feature extractor and matcher included in the Biometric Image Software of the National Institute of Standards and Technology (NIST) [47] can also be applied to enhanced representations of ridge patterns [36]. While not designed for images captured using mobile devices, a minutiae matcher specifically designed for touchless fingerprint images [14] can also be used for fingerprint images acquired using smartphone cameras. Local features other than minutiae points, such as scale-invariant robust features [44], can also be extracted from enhanced representations of ridge patterns.

In recent studies pertaining to learned feature representation, the feature extraction and matching steps have been performed using computational intelligence approaches, with promising results. In particular, it is possible to use scattering networks to extract features and use a random forest classifier to perform the biometric matching [40]. A similar technique using a scattering network and a machine learning classifier is presented in [32]. A competitive coding algorithm and a residual network can also be used in conjunction with a matcher based on the Hamming distance between templates [5].

3.2.5 Quality Assessment

In the quality assessment step, a score value is estimated for each image or local region to represent its ability to be processed by the biometric system with satisfactory results. In the case of selfie fingerprint biometrics, quality estimation can be achieved through three main classes of methods, as follows: (i) estimating the global image quality, (ii) estimating the quality of local regions, and (iii) estimating the focus quality for real-time applications. Methods for estimating the global image quality can be used to reject samples with out-of-focus regions or due to low visibility of the ridge pattern caused by poor illumination. Methods for estimating the quality of local regions can be used to discard poor-quality regions of a sample during the feature extraction process. Methods for estimating the focus quality can be used to implement autofocusing methods specifically designed for selfie fingerprint biometrics. Figure 3.5 shows an example of a fingerprint image in which different regions have different levels of quality.

There are several quality assessment approaches for touchless fingerprint acquisitions pertaining to global image quality estimation. Such methods are typically designed for systems based on either a single camera [13] or multiple cameras [9, 45, 49], but most of them can also be applied to images captured with smartphone cameras. In any case, quality assessment methods trained on images acquired using smartphones can achieve higher accuracy for selfie fingerprint biometrics than methods trained on other types of samples, such as touchless fingerprint images acquired with other kinds of cameras. As an example, the global image quality can be assessed by evaluating the symmetry of the local gradients in the image in combination with a focus estimator [26].

Methods for estimating the local image quality can use sets of features based on the autocorrelation of the fingerprint pattern in the spatial and frequency domains [27] and can also use additional features related to the intensity level of each pixel, the orientation of each local region, and the high-frequency information of the image [28].

Quality assessment methods designed for the real-time selection of correctly focused images are able to run in real time on devices with limited computational power,

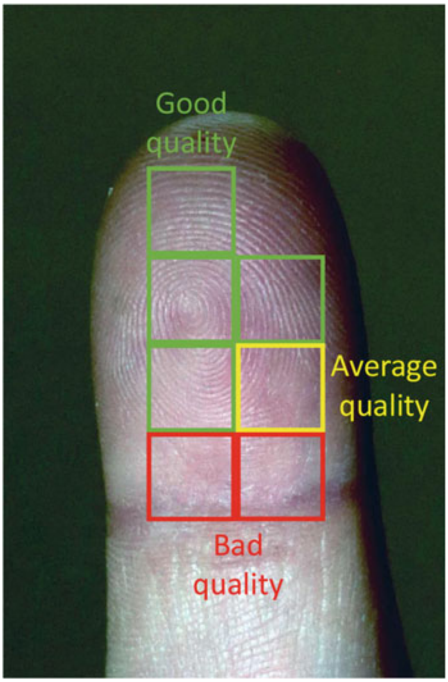


Fig. 3.5 Example of the quality assessment of a touchless fingerprint image captured using a smartphone. The figure shows that different regions offer different levels of visibility of the ridge pattern and are therefore associated with different quality values

such as smartphones. For example, a fast and efficient focus estimator analyzes the density and sharpness of the edges in the image [42].

3.2.6 Liveness Detection

Liveness detection methods aim to distinguish biometric samples of real biometric traits from possible presentation attacks against the sensor [15]. In the case of selfie fingerprint biometrics, this step aims to distinguish real fingers from synthetic artifacts consisting of heterogeneous materials, printouts, and the images shown on electronic devices. Selfie fingerprint biometrics is a recently emerging research field, and there are only a few studies in the literature on methods for liveness detection that are applicable to touchless samples acquired using smartphones.

It is possible to estimate the presence of a spoofing attack based on frame sequences of fingers. In particular, it is possible to analyze the pattern of the reflection of the material while a finger is gradually moving in front of the camera under the light emitted by the integrated LED of the smartphone and then to estimate the edge density of the fingerprint image [41].

Liveness estimation can also be performed on the basis of a single fingerprint image acquired by a smartphone camera. Various texture descriptors (local binary patterns, dense scale-invariant feature transforms, and locally uniform comparison image descriptors) can be used by a support vector machine to distinguish between real and fake fingerprints [43].

Although some methods have not been tested on images captured using a smartphone, there are liveness detection algorithms based on single touchless fingerprint images that could also be evaluated for images acquired using mobile devices. As an example, the method presented in [48] extracts local binary pattern features and computes gray-level co-occurrence matrices to classify each image as real or fake by means of a feedforward neural network classifier.

3.2.7 Mitigation of Nonidealities of Touchless Fingerprint Sensors

This step aims to mitigate the nonidealities of fingerprint images captured with a smartphone (as mentioned in Sect. 3.1). Several methods in the literature include an additional processing step with the purpose of mitigating one or more nonidealities of the captured samples. The methods proposed in the literature can be classified according to their goal: (i) normalizing the fingerprint images to a previously defined resolution, (ii) reducing perspective distortions due to uncontrolled rotations of the finger during acquisition (Fig. 3.6), or (iii) applying surface distortions to increase the compatibility between touchless samples acquired using smartphones and touch-based fingerprint images.

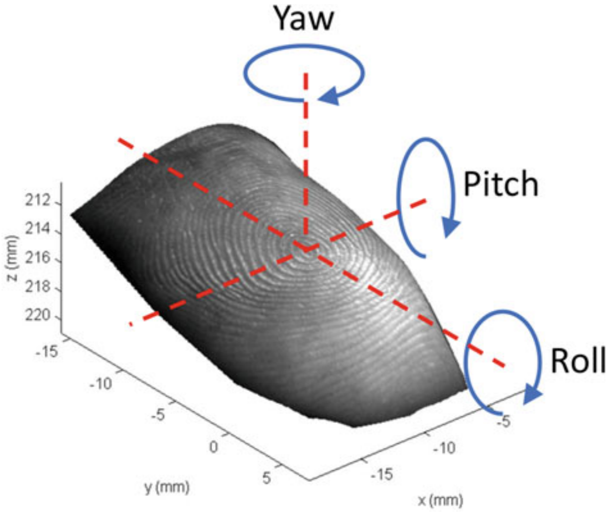


Fig. 3.6 Angles of rotation of a finger

Methods based on normalizing the images aim to mitigate one of the most important nonidealities, namely the uncontrolled resolution of the fingerprint images due to the absence of pins or references for finger positioning, which help to maintain a constant distance between the finger and the camera among different acquisitions [11]. The resulting nonconstant resolution of the samples prevents the direct use of most of the state-of-the-art minutiae-based fingerprint matching methods, such as the NIST BOZORTH3 software [47], which evaluates the Euclidean distances between pairs of minutiae points. To overcome this problem, studies on touchless fingerprint recognition systems normalize the image resolution to approximately 500 pixels per inch by assuming a constant size for each finger [35]. Other studies have normalized the image resolution by assuming that the ridge frequency is constant for each finger [42]. There are also more complex scaling methods that identify the thick valley between the intermediate phalanges and proximal phalanges for scaling the image accordingly [36].

To alleviate the presence of perspective distortions due to uncontrolled rotations of the finger during acquisition, existing methods estimate the rotations of the finger and apply rigid transformations to each fingerprint sample. In particular, the rotation angles of the finger can be estimated using trained neural networks and then used to compute a frontal view image of the fingerprint by rotating a three-dimensional finger model through the estimated angle [10]. Other approaches estimate finger rotations by evaluating the position of the core point and the contour of the finger [26], or apply a correction to the yaw angle of the finger as estimated from its silhouette [42].

Most methods in the literature pertaining to surface distortions require multiview acquisition systems [38] or three-dimensional models [11]. Single fingerprint images acquired using smartphone cameras can also be matched with touchless to touch-based fingerprint images using multisiamese networks [29].

3.3 Performance Analysis

Compared to traditional touch-based systems, touchless fingerprint recognition systems based on less-constrained acquisitions usually exhibit a reduction in accuracy [12] because the lower acquisition constraints result in an increase in the distances between samples belonging to the same user. Among touchless fingerprint recognition systems, selfie fingerprint biometric systems often use the least-constrained acquisition procedures, and therefore, such systems currently achieve lower recognition accuracy compared to fingerprint recognition systems based on more-constrained touchless acquisition devices.

Most studies in the literature use private biometric databases collected by the authors. To the best of our knowledge, there are only two publicly available databases of fingerprint images acquired using smartphones:

- The IIITD SmartPhone Fingerphoto Database v1 (ISPFDv1) [21] is composed of 5100 images captured from 128 fingers using an iPhone 5 with autofocus turned

Table 3.1 Overview of selfie fingerprint biometric recognition methods

References	DB size (Ind./Samp.)	Acquisition	Methodology	Accuracy
[26]	60/1200	Single device, indoor acquisition, uniform background, manual focus assessment	Gabor filtering, minutiae extraction, and matching algorithms for touch-based samples	EER = 4.12% (single device)
[6]	220/1320	Multiple devices, indoor acquisition, uniform background, fixed position, controlled illumination	Commercial software for touch-based samples	EER = 4.66% (single device)
[42]	82/492	Multiple devices, indoor acquisition, uniform background	Median filtering, adaptive binarization procedure, commercial software for touch-based samples	EER = 19.1% (multiple devices)
[27]	100/2100	Multiple devices, indoor and outdoor acquisition, unconstrained background	Commercial software for touch-based samples	EER = 16.9% (all samples, multiple devices); EER = 5.81% (high-quality samples, multiple devices)
[36]	100/1800	Multiple devices, indoor and outdoor acquisition, unconstrained background	Wiener filtering, minutiae extraction, and matching algorithms for touch-based samples	EER = 3.74% (indoor, single device); EER = 2.04% (outdoor, single device, ≈60% FTA)
[44]	50/150	Single device, indoor acquisition, uniform background	Adaptive histogram equalization, SURF features, nearest neighbors	EER = 3.33% (single device)
[40]	128/5100	Single device, indoor and outdoor acquisition, constrained and unconstrained background	Median filtering, histogram equalization, unsharp masking, scattering network, L1 distance	EER = 3.65% (indoor with outdoor matching, single device)
[4]	33/275	Multiple devices, translucent guide on screen, fixed distance, indoor acquisition, uniform background	Band-pass filter, local histogram normalization, commercial software for touch-based samples	FAR = 0.01% @ FRR = 1%
[2]	1500/3000	Single device, indoor acquisition, unconstrained background	Wavelet filtering, phase congruency, commercial software for touch-based samples	EER = 4.8% (single device)
[5]	230/3450	Multiple devices, indoor and outdoor acquisition, unconstrained background, uncontrolled position and illumination	Competitive coding, CNNs, cosine distance	EER = 35.48% (multiple devices)

Notes Ind. = Number of individuals; Samp. = Total number of samples; EER = Equal error rate; FTA = Failure to enroll; FAR = False acceptance rate; FRR = False rejection rate; SURF = Speeded Up Robust Features; CNN = Convolutional neural network on and without any integrated or external illumination source. The images, representing both indoor and outdoor conditions, were collected without the use of pins or references for the finger positioning and have a resolution of 8 megapixels.

- The IIITD Unconstrained Fingerphoto Database (UNFIT) [22] is composed of 3450 images captured from 230 fingers using 45 distinct smartphones and different acquisition software. The images represent both indoor and outdoor conditions, were collected without the use of pins or references for the finger positioning, and have resolutions ranging from 8 to 16 megapixels.

Table 3.1 presents an overview of the fingerprint recognition methods for images acquired using smartphone cameras, describing the size of the dataset considered, the acquisition procedure, the methodology, and the recognition accuracy. This table shows that current biometric systems based on fingerprint images acquired using smartphones can achieve a satisfactory recognition accuracy for many heterogeneous application scenarios. Furthermore, the results obtained when evaluating methods using sets of images collected under both indoor and outdoor conditions are worse than those achieved when evaluating methods on images collected only under indoor conditions. Similarly, the results obtained for images acquired using multiple different smartphones are inferior to those achieved for images acquired using a single device.

3.4 Conclusions

This chapter presents a review of methodologies for selfie fingerprint biometrics. The existing methods for fingerprint recognition are described, analyzing every step of the biometric recognition process. The performance of the state-of-the-art methods is also compared and analyzed.

State-of-the-art methods enable the acquisition and processing of images of multiple fingerprints with uncontrolled finger positioning and uncontrolled background and illumination conditions. They may use enhancing algorithms and standard minutiae-based recognition techniques or may be based on dedicated feature extractors and matchers. There are also methods for quality estimation, liveness detection, resolution normalization, and the mitigation of perspective distortions as well as techniques for improving the compatibility between touchless and touch-based samples.

Currently, selfie fingerprint biometrics can achieve satisfactory accuracy for a wide variety of identity verification applications. However, these systems are less accurate than traditional touch-based fingerprint recognition technologies. This is because smartphone-based systems use samples acquired under less-constrained conditions, which present additional challenges with respect to touch-based fingerprint images. Furthermore, the results reported in the literature show that there are two main aspects of the acquisition process that contribute to reducing the recognition accuracy: (i) acquiring images using heterogeneous smartphones and (ii) performing outdoor acquisition with uncontrolled illumination and background conditions.

To improve the usability of selfie fingerprint biometric techniques, current research trends are oriented toward further lowering the acquisition constraints by considering multifingerprint samples acquired in different outdoor scenarios, with uncontrolled backgrounds, illumination conditions, and finger positioning. At the same time, researchers are focusing on improving the recognition accuracy by designing novel enhancement techniques, more efficient feature extraction and matching algorithms such as methods based on deep learning and convolutional neural networks.

Acknowledgements This work was supported in part by the Italian Ministry of Research as part of the PRIN 2015 project COSMOS (201548C5NT).

References

1. Apple: Face ID. http://support.apple.com/en-us/HT208108
2. Birajadar P, Gupta S, Shirvalkar P, Patidar V, Sharma U, Naik A, Gadre V (2016) Touch-less fingerphoto feature extraction, analysis and matching using monogenic wavelets. In: Proceeding of the 2016 international conference on signal and information processing (IConSIP), pp 1–6
3. Blanco-Gonzalo R, Sanchez-Reillo R (2009) Biometrics on mobile devices. In: Li SZ, Jain AK (eds) Encyclopedia of biometrics. Springer, US, Boston, MA, pp 1–8
4. Carney LA, Kane J, Mather JF, Othman A, Simpson AG, Tavanai A, Tyson RA, Xue Y (2017) A multi-finger touchless fingerprinting system: mobile fingerphoto and legacy database interoperability. Proceeding of the 2017 4th international conference on biomedical and bioinformatics engineering (ICBBE). ACM, New York, NY, USA, pp 139–147
5. Chopra S, Malhotra A, Vatsa M, Singh R (2018) Unconstrained fingerphoto database. In: Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR) workshops
6. Derawi MO, Yang B, Busch C (2012) Fingerprint recognition with embedded cameras on mobile phones. In: Prasad R, Farkas K, Schmidt AU, Lioy A, Russello G, Luccio FL (eds) Security and privacy in mobile information and communication systems. Springer, Berlin Heidelberg, Berlin, Heidelberg, pp 136–147
7. Donida Labati R, Genovese A, Muñoz E, Piuri V, Scotti F (2017) A novel pore extraction method for heterogeneous fingerprint images using Convolutional Neural Networks. Pattern Recognition Letters
8. Donida Labati R, Genovese A, Piuri V, Scotti F (2010) Measurement of the principal singular point in contact and contactless fingerprint images by using computational intelligence techniques. In: Proc. of the IEEE international conference on computational intelligence for measurement systems and applications, pp 18–23 (2010)
9. Donida Labati R, Genovese A, Piuri V, Scotti F (2012) Quality measurement of unwrapped three-dimensional fingerprints: a neural networks approach. Proceeding of the 2012 IEEE-INNS international joint conference on neural networks (IJCNN). Brisbane, Australia, pp 1123–1130
10. Donida Labati R, Genovese A, Piuri V, Scotti F (2013) Contactless fingerprint recognition: a neural approach for perspective and rotation effects reduction. In: Proceeding of the IEEE workshop on computational intelligence in biometrics and identity management (CIBIM). Singapore, pp 22–30
11. Donida Labati R, Genovese A, Piuri V, Scotti F (2014) Touchless fingerprint biometrics: a survey on 2D and 3D technologies. J Internet Technol 15(3):325–332

12. 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
13. Donida Labati R, Piuri V, Scotti F (2010) Neural-based quality measurement of fingerprint images in contactless biometric systems. In: Proceeding of the 2010 IEEE-INNS international joint conference on neural networks (IJCNN). Barcelona, Spain, pp 1–8
14. Donida Labati R, Piuri V, Scotti F (2011) A neural-based minutiae pair identification method for touch-less fingerprint images. In: Proceeding of the IEEE workshop on computational intelligence in biometrics and identity management (CIBIM), pp 96–102
15. Donida Labati R, Piuri V, Scotti F (2012) Biometric privacy protection: guidelines and technologies. In: Obaidat MS, Sevillano J, Joaquim F (eds) Communications in computer and information science, vol 314. Springer, pp 3–19
16. Donida Labati R, Piuri V, Scotti F (2015) Touchless fingerprint biometrics. Series in security, Privacy and Trust. CRC Press
17. 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 Biom 5(1):28–36
18. Goodfellow I, Bengio Y, Courville A (2016) Deep Learning. MIT Press
19. Hiew BY, Teoh ABJ, Pang YH (2007) Touch-less fingerprint recognition system. In: Proceeding of the 2007 IEEE workshop on automatic identification advanced technologies, pp 24–29
20. Hiew BY, Teoh ABJ, Yin OS (2010) A secure digital camera based fingerprint verification system. J Vis Commun Image Represent 21(3):219–231
21. IIIT Delhi: IIITD SmartPhone Fingerphoto Database v1 (ISPFDv1). http://iab-rubric.org/resources/spfd.html
22. IIIT Delhi: Unconstrained Fingerphoto Database (UNFIT). http://iab-rubric.org/resources/UNFIT.html
23. Kakumanu P, Makrogiannis S, Bourbakis N (2007) A survey of skin-color modeling and detection methods. Pattern Recognit 40(3):1106–1122
24. Kumar A, Kwong C (2015) Towards contactless, low-cost and accurate 3D fingerprint identification. IEEE Trans Pattern Anal Mach Intell 37(3):681–696
25. Lee C, Lee S, Kim J, Kim SJ (2005) Preprocessing of a fingerprint image captured with a mobile camera. In: Zhang D, Jain AK (eds) Advances in biometrics. Springer, Berlin Heidelberg, Berlin, Heidelberg, pp 348–355
26. Lee D, Choi K, Choi H, Kim J Recognizable-image selection for fingerprint recognition with a mobile-device camera. IEEE Trans Syst, Man, Cybern, Part B (Cybernetics) 38(1):233–243 (2008)
27. Li G, Yang B, Busch C (2013) Autocorrelation and DCT based quality metrics for fingerprint samples generated by smartphones. In: Proceeding of the 2013 18th international conference on digital signal processing (DSP), pp 1–5
28. Li G, Yang B, Olsen MA, Busch C (2013) Quality assessment for fingerprints collected by smartphone cameras. In: Proceeding of the 2013 IEEE conference on computer vision and pattern recognition (CVPR) workshops, pp 146–153
29. Lin C, Kumar A (2017) Multi-siamese networks to accurately match contactless to contact-based fingerprint images. In: Proceeding of the IEEE international joint conference on biometrics (IJCB), pp 277–285
30. Liu F, Zhang D, Song C, Lu G (2013) Touchless multiview fingerprint acquisition and mosaicking. IEEE Trans Instrum Meas 62(9):2492–2502
31. Liu X, Pedersen M, Charrier C, Cheikh FA, Bours P (2016) An improved 3-step contactless fingerprint image enhancement approach for minutiae detection. In: Proceeding of the 2016 6th European workshop on visual information processing (EUVIP), pp 1–6
32. Malhotra A, Sankaran A, Mittal A, Vatsa M, Singh R (2017) Fingerphoto authentication using smartphone camera captured under varying environmental conditions. In: Marsico MD, Nappi M, Proenca H (eds) Human recognition in unconstrained environments. Academic Press, pp 119–144

33. Maltoni D, Maio D, Jain AK, Prabhakar S (2009) Handbook of fingerprint recognition, 2nd edn. Springer Publishing Company, Incorporated
34. Neurotechnology: VeriFinger SDK. http://www.neurotechnology.com/verifinger.html
35. Piuri V, Scotti F (2008) Fingerprint biometrics via low-cost sensors and webcams. In: Proceeding of the 2008 IEEE international conference on biometrics: Theory, Applications and Systems (BTAS). Washington, D.C., USA pp 1–6
36. Raghavendra R, Busch C, Yang B (2013) Scaling-robust fingerprint verification with smartphone camera in real-life scenarios. In: Proceeding of the 2013 IEEE 6th International Conference on Biometrics: theory, applications and systems (BTAS), pp 1–8 (2013)
37. Ross A, Nandakumar K, Jain AK (2008) Introduction to multibiometrics. In: Jain AK, Flynn P, Ross AA (eds) Handbook of Biometrics. Springer, US, Boston, MA, pp 271–292
38. Salum P, Sandoval D, Zaghetto A, Macchiavello B, Zaghetto C (2017) Touchless-to-touch fingerprint systems compatibility method. In: Proceeding of the 2017 IEEE international conference on image processing (ICIP), pp 3550–3554
39. Samsung: Iris scan. http://www.samsung.com/in/smartphones/galaxy-s8/security/
40. Sankaran A, Malhotra A, Mittal A, Vatsa M, Singh R (2015) On smartphone camera based fingerphoto authentication. In: Proceeding of the 2015 IEEE 7th international conference on biometrics theory, applications and systems (BTAS), pp 1–7
41. Stein C, Bouatou V, Busch C (2013) Video-based fingerphoto recognition with anti-spoofing techniques with smartphone cameras. In: Proceeding of the international conference of the BIOSIG special interest group (BIOSIG), pp 1–12
42. Stein C, Nickel C, Busch C (2012) Fingerphoto recognition with smartphone cameras. In: Proceeding of the 2012 international conference of biometrics special interest group (BIOSIG), pp 1–12 (2012)
43. Taneja A, Tayal A, Malhorta A, Sankaran A, Vatsa M, Singh R (2016) Fingerphoto spoofing in mobile devices: a preliminary study. In: Proceeding of the 2016 IEEE 8th International conference on biometrics theory, applications and systems (BTAS), pp 1–7
44. Tiwari K, Gupta P (2015) A touch-less fingerphoto recognition system for mobile hand-held devices. In: Proceeding of the 2015 international conference on biometrics (ICB), pp 151–156
45. Wang Y, Hao Q, Fatehpuria A, Hassebrook LG, Lau DL (2009) Data acquisition and quality analysis of 3-dimensional fingerprints. In: Proceeding of the 2009 1st IEEE internatioanal conference on biometrics, identity and security (BIdS), pp 1–9
46. Wang Y, Hassebrook LG (2010) Lau DL (2010) Data acquisition and processing of 3-D fingerprints. IEEE Trans Inf Forensics Secur 5(4):750–760
47. Watson CI, Garris MD, Tabassi E, Wilson CL, Mccabe RM, Janet S, Ko K (2007) User's guide to NIST biometric image software (NBIS) (2007)
48. Zaghetto C, Mendelson M, Zaghetto A, dB Vidal F (2017) Liveness detection on touchless fingerprint devices using texture descriptors and artificial neural networks. In: Proceeding of the IEEE internatioanal joint conference on biometrics (IJCB), pp 406–412 (2017)
49. Zaghetto C, Zaghetto A, dB Vidal F, Aguiar LHM (2015) Touchless multiview fingerprint quality assessment: rotational bad-positioning detection using artificial neural networks. In: Proceeding of the 2015 internatonal confirence on biometrics (ICB), pp 394–399

Chapter 4
MICHE Competitions: A Realistic Experience with Uncontrolled Eye Region Acquisition

Silvio Barra, Maria De Marsico, Hugo Proença and Michele Nappi

Abstract People struggle every day with authentication to access a protected service or location, or simply aimed at protecting one's own devices. This spurs a growing demand for self-handled authentication strategies. The increasing number of remote services of various kinds corresponds to an increasing number of passwords to use and remember, and also to the growth of the password theft risk, due to the increasing value of the protected resources. The other core element in present authentication scenarios is the ubiquity of mobile equipment. Smartphones add a "whatever" dimension to the possible uses of the mobile devices whenever and wherever that include storing/transferring multimedia information, often personal and often sensitive. Biometrics can both enforce and simplify authentication in controlled environments. Mobile biometrics in uncontrolled settings, where there is no operator to guide the capture of a "good-quality" sample on a mobile device, is the new frontier for secure use of data and services. The iris is among the best candidates for biometric recognition. It is extremely discriminative: Right and left irises of the same person are so different to hinder a correct matching, because randotypic elements largely overcome genotypic ones in individual development. However, self-acquired samples often suffer from poor quality, due, e.g., to reflections, motion blurring, out of focus, or bad image framing. Mobile setting and especially the inherent problems related to uncontrolled iris image acquisition are addressed in the two challenges of the MICHE project, whose results are the core topic of this chapter.

S. Barra
University of Cagliari, Cagliari, Italy
e-mail: silvio.barra@unica.it

M. De Marsico (✉)
Sapienza University of Rome, Rome, Italy
e-mail: demarsico@di.uniroma1.it

H. Proença
Universidade da Beira Interior, Covilhã, Portugal
e-mail: hugomcp@di.ubi.pt

M. Nappi
University of Salerno, Fisciano, Italy
e-mail: mnappi@unisa.it

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision
and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_4

4.1 Introduction

Any user in the world that has to access a protected service or location, or that simply wants to protect its owned devices, has to struggle with assuring a secure access to them. This is a first aspect that characterizes self-handled authentication strategies. Actually, the use of special signs, objects, or passphrases goes back to the very origins of human communities. Watchwords asked by sentinels, or the five-pointed pentagon tattooed on the palm of members of the Pythagorean school, are examples of a kind of authentication often seen in the literature. The first attempt to use computer support for authentication is represented by passwords that first appeared at the Massachusetts Institute of Technology in the mid-1960s. There, a massive compatible time-sharing computer (CTSS) was used to pioneer many of the milestones of computing, including password-based authentication. In those times, a single password was sufficient to access one's virtual space and files, which after all were the only resources to protect. Afterward and beyond any forecasting, computers massively entered everyday life, with Internet allowing the creation of an increasing number of remote services of various kinds. This has caused both the corresponding increase of the number of passwords to use and also the growth of the password theft risk, due to the increasing value of the protected resources. More and more complex and non-trivial passwords must be used. However, the more they are difficult to crack, the more they are difficult to remember. The possible alternative or addition represented by possession of physical objects (e.g., keys and cards) does not solve these problems. Rather, the need to keep the physical object always available when needed, and the possibility that it can be lost or stolen, may make things even worse for the users. In this awkward scenario, biometric authentication, though not being invincible, seems to provide a more "natural" alternative. The users can just exploit what they are or the way they behave to be recognized and granted privileges.

The other core element in present authentication scenarios is that mobile equipment is ubiquitous nowadays. Smartphones substituted old cellular phones that in turn had replaced traditional landlines. The possibility to communicate almost wherever and whenever represents a characterizing aspect of the still ongoing technological revolution. However, the whatever dimension, that allows the new communication devices and protocols, is even more disrupting. The uses of present smart mobile devices include storing/transferring in real time almost any kind of multimedia information. Such data is often personal and often sensitive. The exchange of sensitive information requires a twofold approach to address increasing security needs: It is necessary both to reliably identify the owner before the use of the device and to reliably identify the user of a remote service at the moment the device connects to it. Biometrics can both enforce and make authentication simpler in conventional controlled environments. The next step is to move biometrics in uncontrolled settings, where there is no operator to guide the capture of a "good-quality" sample and on mobile devices. Mobile biometric recognition is the new frontier for secure use of data and services.

It is interesting to remind the basic principles and issues that characterize biometric authentication. The paper by Clarke published in 1994 [20] is among the earliest ones devoting specific attention to biometric recognition. Available means to achieve formal identification of individuals are classified as: (1) ways to merely distinguish among individuals—names and codes; (2) ways to verify individual identity—knowledge-based identification and token-based identification; and, finally, (3) biometrics that can be used for both verification and identification. In this classification, the term "biometrics" refers to identification techniques relying on some physical and difficult-to-alienate characteristic. Of course, they require suitable measurements and matching strategies. Clarke further sketches a first taxonomy of biometric traits [20]: (1) those based on appearance that include the usual elements reported in any identity document, such as height, weight, color of skin, hair and eyes, visible markings, gender, race, facial hair, glasses that are supported by photographs; (2) those based on (social) behavior, including body signals, voice characteristics, speech style, visible handicaps that are supported by video (or audio) recordings; (3) those based on biodynamics, including the way of signing and keystroke dynamics that require specific capture strategies; (4) those based on natural physiography, including skull measures, teeth and skeletal injuries, fingerprint sets and handprints, retinal scans, vein patterns, hand geometry, and DNA; and (5) those based on imposed physical characteristics, including collars, bracelets, microchips, and transponders. The paper by Jain et al. [37] simplifies this classification into two broad classes, namely physical or behavioral traits, that are still used at present as reference. The paper further elaborates on Clarke's human identifiers to list the properties of a biometric trait. They are the well-known universality, uniqueness, collectability, performance, and circumvention. In [36], biometric traits are further classified as either supporting unique identification (hard traits, e.g., face or fingerprints) or providing information lacking sufficient distinctiveness and/or permanence to differentiate any two individuals (soft traits, e.g., demographic traits and most behavioral traits).

Notwithstanding the optimistic premises, the kind of interaction required from the biometric recognition systems may cause troubles to non-expert users, especially in unattended scenarios where no operator is there to assist during the task. In general, authentication systems are often difficult to use. Quoting from a paper published in 2001 by Sasse et al. [63]: "The security research community has recently recognized that user behavior plays a part in many security failures, and it has become common to refer to users as the 'weakest link in the security chain'. We argue that simply blaming users will not lead to more effective security systems." In 2000, Nielsen [46] assumes that "in the future, security will improve through biological [biometric] verification mechanisms, such as fingerprint recognition or retina scanning" and yet also alerts that "it will take time for this infrastructure to be built (and fingerprint systems will not work for some people)." The conclusion in [63] is even more skeptical: "Biometric systems may be a good fit for some user–tasks–context configurations, but not all of them." Concerns raised in 2004 [48] and related to the acquisition step are unfortunately still valid, as researchers dealing with biometric recognition know very well. Fingerprint readers may suffer from dirt, bad framing, a different pressure and motion; face recognition systems are affected by PIE (pose, illumination, expression) distortions and also by aging of the subject. Iris scanners may suffer from the bad alignment of the eye with the camera lens (e.g., off-axis).

These problems become dramatically critical when dealing with mobile biometrics. In this case, more problems rise because of the unattended acquisition, since the user may not be able to capture a good sample and further be unaware of what capturing a "good sample" means in the different cases. The 2007 work by Sasse [62] proposes an apparently obvious solution: "Biometric systems should have user-friendly, intuitive interfaces that guide users in presenting necessary traits." However, reliable use of selfie biometrics is still an open problem. Mobile biometric recognition is continuously increasing its popularity, thanks to the possibility of exploiting personal and/or wearable devices equipped with more and more accurate sensors. Mobile equipment is ubiquitous nowadays and allows capturing biometric traits anytime in any place, by incorporating all necessary hardware equipment and software applications for capturing and processing biometric data. However, the capture phase still poses crucial problems. This dichotomy (Fig. 4.1) inspired Mobile Iris CHallenge Evaluation (MICHE) project.

The chapter develops as follows. Section 4.2 summarizes the main concepts related to iris recognition and how MICHE challenges are positioned with respect to the past and present research scenarios. Section 4.3 briefly describes the challenge setup with its two separate phases and the dataset used as benchmark for evaluating participating approaches. Section 4.4 deals with the first MICHE-I challenge, focused on iris segmentation. Section 4.5 presents the results of the following MICHE-II challenge, focused on iris recognition.

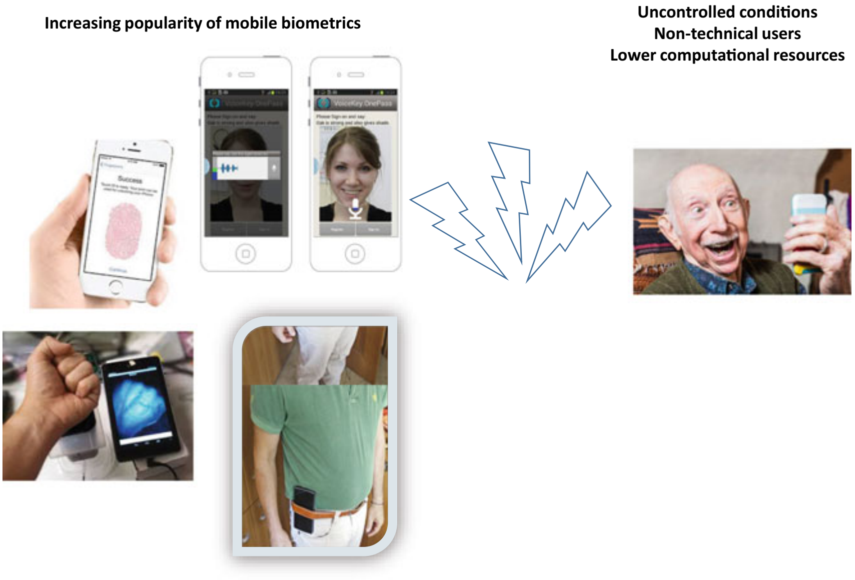

Increasing popularity of mobile biometrics
Uncontrolled conditions
Non-technical users
Lower computational resources

Fig. 4.1 Increasing popularity of mobile biometrics versus increasing use by non-technical users

4.2 Iris Recognition and MICHE Challenges

The iris is the circular structure in the central part of the eye that determines what is popularly defined as the *eye color*. From a functional point of view, it has a muscular nature that is responsible for controlling the diameter and size of the pupil (the inner black disk, actually a hole) and therefore the amount of light reaching the retina. From an optical point of view and comparing the eye to a camera, the pupil represents the aperture, while the iris has the role of the diaphragm. From the biometric recognition point of view and of the involved processing steps, it is worth reminding which are the most relevant external visible structures that contribute to characterize a human iris. The pupillary zone is the most internal part of the iris, whose edges mark the pupil boundary. These edges are well visible in light color eyes, while it may be difficult to distinguish them in very dark eyes. The latter is one of the problems to be addressed during iris region segmentation in visible light. Proceeding toward the external iris border, the collarette is a very thick region that separates the pupillary region from the ciliary zone. In this region, the sphincter muscle and dilator muscle regulate the pupil dilation. It is relatively easy to identify this region in eyes which are not too dark, since it is made up of radial ridges extending from the periphery to the pupillary zone. The ciliary zone extends up to meet the sclera. The overall iris structure is characterized by both regularities, represented by radial furrows, and singularities, represented by crypts and possible lighter/darker spots (Fig. 4.2).

The iris is among the best candidates for biometric recognition. It is extremely discriminative: Right and left irises of the same person are so different to hinder a correct matching. This is due to the fact that randotypic elements largely overcome genotypic ones in individual development. In other words, contrarily to, e.g., face, the genetic baggage has very little influence on the iris makeup process. Its small size makes related image processing quite fast with respect to face, and its peculiarities make it very difficult to spoof an iris template. The most used kind of iris codes,

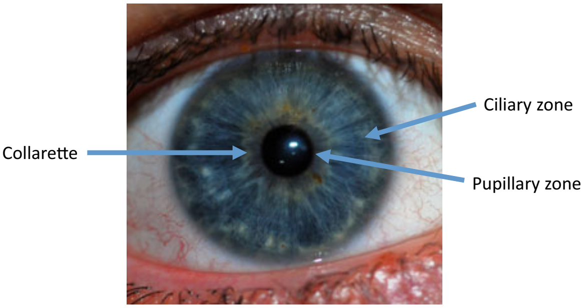


Fig. 4.2 Most relevant regions of iris images, for biometric recognition purposes. The collarette divides the pupillary and ciliary zones, as is particularly visible in light-pigmented irises. Image adapted from the original image by JDrewes - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=3117810

devised by Daugman [22], is among the less expensive templates from the storage point of view, and the acquisition is little intrusive. For all these reasons, the iris is a natural candidate for mobile biometric recognition.

Research results regarding related techniques have quickly progressed from the pioneering work by Daugman [22] and Wildes [66], mostly pertaining controlled settings and near-infrared (NIR) capture settings, to the use of deep learning [41], with the most recent Noisy Iris Challenge Evaluation (NICE) addressing iris recognition in less controlled settings [50, 52]. In addition to evolving iris image processing techniques, the periocular region is deserving increasing attention, to either complement iris recognition or to be used with images with too low resolution. Among the most recent works, Reddy et al. [59] propose Ocular-Net, a convolution neural network (CNN) model, using six registered overlapping patches from the ocular and periocular region; these are extracted to train a small CNN for each patch named PatchCNN to extract feature descriptors. As the proposed method is a patch-based technique, one can extract features based on the availability of the region in the eye image. The proposed Ocular-Net with 1.5 M parameters obtained comparative performance with popular ResNet-50 model which has 23.4 M parameters.

Mobile setting and especially the inherent problems related to uncontrolled acquisition are addressed in the two challenges of the MICHE project [24, 25] whose results are the core topic of this chapter. Whatever the context, the iris recognition workflow follows the same processing steps, which are typical of any object detection/recognition procedure. The ease of localizing the eyes within the faces, and the characteristic annular shape of the iris, should facilitate a reliable and accurate detection of this anatomical element and the creation of a suitable representation. This especially holds when NIR capturing is used, since reflections and illumination variations have little influence on images, and a controlled acquisition guarantees a correct position with respect to the camera. On the contrary, when capture is carried out in visible light (VL), images usually contain precious chromatic features than NIR images, but they are also much more seriously affected by many noisy artifacts produced by light sources and reflections [57], and their processing suffers from possible dark pigmentation. Therefore, the first difficulties soon arise when attempting to detect and segment the iris. Of course, a poor segmentation compromises all the following steps, since feature extraction would be possibly carried out on non-iris regions, while the complete set of (possibly unconnected) iris regions would not be correctly identified. Iris segmentation was the focus of the first MICHE challenge, aimed at assessing the accuracy of the candidate algorithms. It is worth noticing that segmentation not only identifies the useful iris region, but also usually produces a segmentation mask to be used during matching to leave out non-iris patches. The step following segmentation is iris sample normalization. In most approaches, this does entail not only reducing iris images to a common size, but also computing a polar representation facilitating the following processing. The most used technique to obtain this is the rubber sheet model introduced by Daugman [21]. The same procedure is applied to the segmentation mask. Afterward, different approaches extract and match different features, related either to the regular patterns that can be identified or to possible singularities, either local or global [19]. Feature extraction

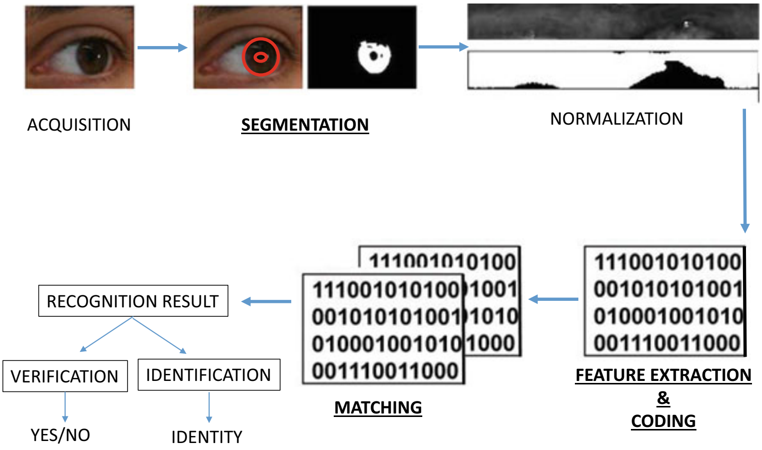


Fig. 4.3 Main phases of iris recognition processing chain (bold font underlines the phases addressed by MICHE challenges)

and recognition are the focus of the second MICHE challenge. Figure 4.3 shows the typical steps in iris processing and recognition and points out those addressed by MICHE. Of course, the kind of the final result depends on the entailed recognition modality, either verification (1:1 matching) or identification (1:N matching).

4.3 Challenge Setup and MICHE Dataset

As anticipated above, the Noisy Iris Challenge Evaluation I (NICE I) addressed the problem of matching images captured in unconstrained conditions. The iris dataset used as benchmark, namely UBIRIS.v2 [53], was captured in the visible wavelength (VW), at-a-distance (4–8 m), and on-the-move. The results of the challenge confirm how VW and uncontrolled conditions together dramatically affect recognition performance. Similar conclusions result from the following NICE II contest [52]. Going further along the line of increasingly challenging conditions, MICHE project addresses a further problem. While UBIRIS datasets were acquired by high-resolution cameras, MICHE dataset, as we will better detail in the following, only uses built-in cameras of different smartphones that at the moment of capture produced images of undoubtedly lower quality than UBIRIS ones. MICHE challenges followed the same NICE schema: a first one focusing on iris segmentation and using as benchmark MICHE dataset, and a second one focusing on feature extraction and matching, carried out with an extended version of the dataset, and using as a common segmentation tool the best method resulted from the first challenge.

Why was a new dataset required? The Chinese Academy of Sciences was a pioneer in collecting the first publicly available datasets dealing with iris images, continuously updated from CASIA-IrisV1 to CASIA-IrisV4 since 2002 [43, 64]. Its images are either collected under NIR or are synthesized. Therefore, until NIR sensors will truly spread on mobile devices, these datasets cannot be used to assess everyday life iris processing on mobiles. Similar considerations hold for benchmarks used for ICE competitions [68, 69]. On the contrary, UBIRIS datasets, available from SOCIA Lab at University of Beira Interior (Portugal), were captured in visible light and uncontrolled conditions. However, they have a much better resolution than average mobile sensors. Figure 4.4 shows some details of the last CASIA versions, while Fig. 4.5 shows a sample from a ICE competition and a sample from UBIRIS, both from a left eye. Looking at the two samples, it is easy to understand the difference between the two addressed contexts.

The aim of MICHE was to assess the real feasibility of iris recognition when images are captured in visible light, by "normal" user-level mobile devices and by "normal" (not necessarily technical) users in uncontrolled/unattended conditions, and when cross-device matching can be needed (Fig. 4.6).

MICHE challenges provided as a benchmark a dataset reflecting this specific setup. Before continuing, it is worth pointing out two symmetrical considerations. The accuracy of capture/quality of the captured image may be enhanced due to the usually short distance (not more than the length of a normal human arm) and to the user natural attitude to have a frontal pose while taking a selfie. In addition,

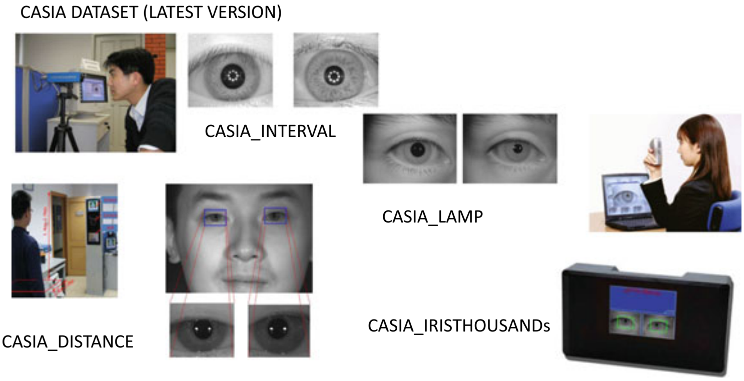


Fig. 4.4 Last versions of CASIA datasets

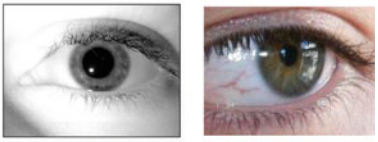


Fig. 4.5 A sample from ICE (left) and one from UBIRIS (right)

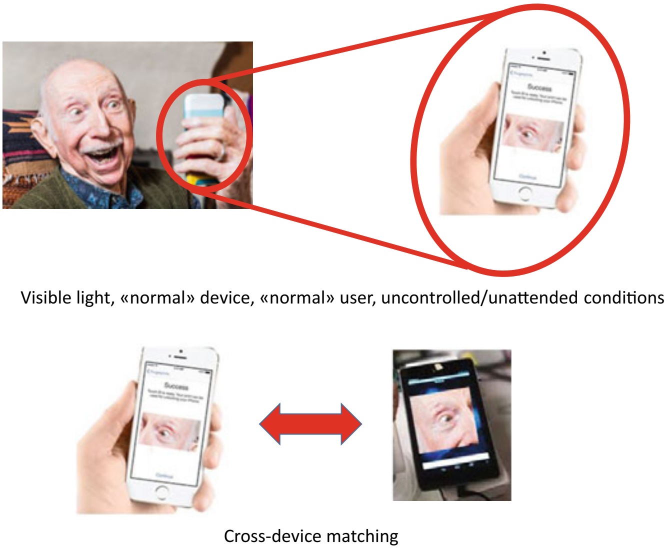

Visible light, «normal» device, «normal» user, uncontrolled/unattended conditions
Cross-device matching

Fig. 4.6 Cohesive perspective of the MICHE operational context

in this case we are considering collaborative users that have all interest in being recognized. The reverse of the medal is that the quality of the captured image can suffer from possible lower resolution of the mobile device camera, from motion blur and illumination distortions, and from incorrect image framing that can be all caused by either/both kinds of device, the possible lack of technical experience of the user, and by the lack of control on user capture operation. Addressing these problems requires more robust detection/segmentation and matching procedures. It is worth pointing out again that the performance of the matching can be dramatically affected by the quality of the segmentation. This is the reason for having all participants to the second part of the challenge to all use the same segmentation: They have a common starting point so that it is possible to evaluate the addition of feature extraction/matching in a fair way.

The composition of the dataset used for MICHE reflects the use of different mobile devices for the acquisition and a realistic simulation of the acquisition process including different sources of distortion/noise. The data was captured across several acquisition sessions separated in time, to get a realistic amount of intra-class variations. All images were annotated with metadata useful to carry out demographics as well as well as device-based analysis. In order to reproduce a real-world setting, the subjects involved in experiments were given no special instructions but were rather advised to take a selfie of their eye as they would do if asked in a real situation. For instance, subjects usually wearing eyeglasses could either remove or keep them. The self-images of their iris were acquired by normally holding the mobile device. For each session, a minimum of four shots for each camera (a device could possibly have two) and acquisition mode (either indoor or outdoor) was requested. Indoor acquisition was affected by various sources of artificial light, sometimes combined with natural light sources. Outdoor acquisition exploited natural light only. Only one iris per subject was acquired. The three kinds of devices used for data acquisition purposes (both smartphones and tablets) that (at the time!) were representative of the current top market category can represent at present medium-level devices and are listed by increasing camera resolution:

- Galaxy Tablet II (GT2)
  - Operating system: Google's Android
  - Posterior camera: N/A
  - Anterior camera: 0.3 megapixels
- iPhone5 (IP5)
  - Operating system: Apple iOS
  - Posterior camera: iSight with 8 megapixels (72 dpi)
  - Anterior camera: FaceTime HD camera with 1.2 megapixels (72 dpi)
- Galaxy Samsung IV (GS4)
  - Operating system: Google's Android
  - Posterior camera: CMOS with 13 megapixels (72 dpi)
  - Anterior camera: CMOS with 2 megapixels (72 dpi).

It is interesting to point out that it is possible to identify three groups of images at three different resolutions (1536 × 2048 for iPhone5, 2322 × 4128 for Galaxy S4, and 640 × 480 for the tablet). Examples are shown in Fig. 4.7. This is a further challenge for cross-device matching.

Sources of noise affecting the MICHE dataset images include all those that can be present in real-world unattended settings. Different kinds of reflexes are among the most frequent ones and can be caused by either artificial light or natural light sources, as well as by people or objects in the scene. Out of focus and blur can be either due to an incorrect capture operation or due to involuntary movements of the hand and/or of the head and/or of the eye during selfie capturing. Part of the region of interest may be occluded by eyelids, eyeglasses, eyelashes, hair, or shadows. The device itself may introduce artifacts due to low resolution or sensor defects, or it may present different color dominants. Further problems are raised by off-axis gaze and variable illumination. Actually, Fig. 4.8 shows that these factors can also affect images in UBIRIS.v2 (http://nice2.di.ubi.pt/). However, comparison of Figs. 4.7 and 4.8 is useful to further point out the features of MICHE images. Most of all, due to lack of precise framing and to different capture distances, it is possible to obtain either well-centered eyes or half faces or partial eye images. Of course, this causes

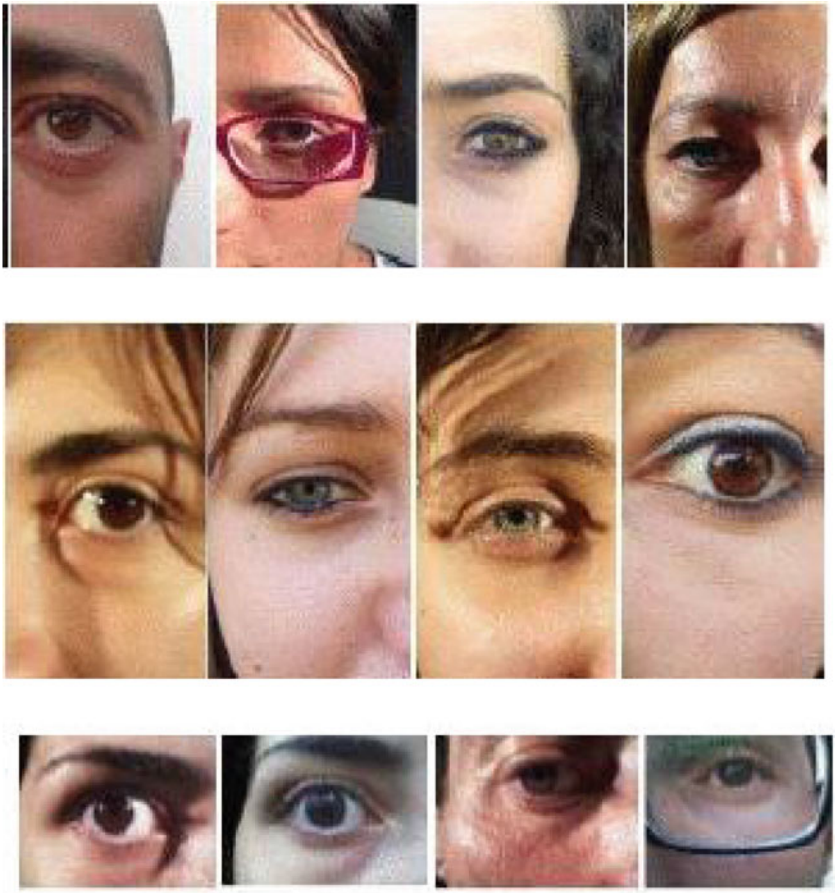


Fig. 4.7 Examples of MICHE images. From top to bottom, images were taken by iPhone5, by Galaxy S4, and by Galaxy Tablet II devices

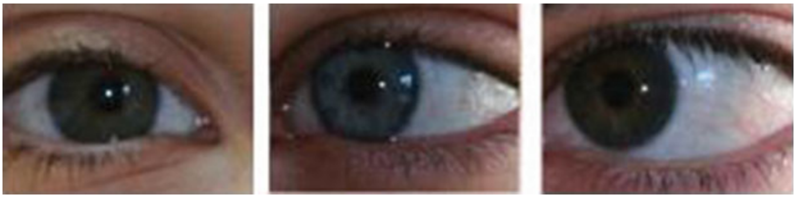


Fig. 4.8 Examples of UBIRIS images

a different position, size, and sharpness of the region of interest, i.e., the region useful for iris recognition. This happens because capture can happen from a very close distance up to a distance equal to the arm length, even if in general the users maintain an average capture distance that is in the middle. This means that more robust eye localization techniques are required as a first processing step, but at the same time it is possible to use, when present, useful information from the periocular region.

The dataset is annotated by metadata through a reference XML file for each iris image. The information recorded regards image acquisition (e.g., device characteristics, distance from the device, outdoor/indoor indication) and archiving (e.g., file name and file type), subject demographics, and the conditions under which the image was acquired. At present, MICHE dataset contains images from 75 different subjects, acquired in two sessions separated in time (1–9 months apart) with 1297 images from GS4, 1262 images from IP5, and 632 images from GT2. The dataset also contains a MICHE Fake and a MICHE Video subsets, to test presentation attack detection (PAD) approaches and recognition from dynamic data. More details on the dataset can be found in [26]. Table 4.1 summarizes the difference in terms of possible distortions between controlled and uncontrolled acquisition. It is worth pointing out that "controlled" in this context means assisted by an expert operator that can appreciate the possible defects in the obtained image and ask to repeat the acquisition until a satisfactory visual quality is achieved. Low resolution can be considered as a possibly common problem for the two settings, depending on the acquisition device. However, in uncontrolled conditions it can add to possible other distortions and can be more frequent in mobile capture.

4.4 MICHE-I Challenge: Iris Segmentation

The participants to MICHE-I challenge had the above described dataset as a common benchmark. The aim of the challenge and of the analyses carried out on its results was to explore both image covariates that are likely to cause a decrease in the performance levels of the compared algorithms and the further effect of cross-device operations. Segmentation was the only focus of the challenge; however, participants had the possibility to also integrate their proposal with a recognition module. This allowed to test both the original approaches and possible recombinations of segmentation (S) and recognition (R) modules.

The analysis of results went beyond the evaluation/identification of the best approaches, but also focused on the image features/distortions that can mostly positively/negatively affect the final recognition performance and on interoperability issues caused by the use of different devices in enrollment vs. testing phases. Having more recognition modules available, it was also possible to investigate multi-classifier strategies, to complement the strengths of more different approaches. The fusion was

Table 4.1 Summary of the data degradation factors affecting image acquisition in controlled/uncontrolled conditions

	Motion blur	Out of focus	Reflections	Occlusions (eyelids/ eyelashes)	Off-axis	Closed/ partially closed eye	Partial eye region (bad framing)	Extended periocular region (bad framing)	Low resolution
Controlled	No	No	Yes	Yes	No	No	No	No	Yes/No
Uncontrolled	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes/No

carried out at score level either by the popular simple sum or by a weighted sum policy assigning higher weights to methods achieving a lower equal error rate (EER) in a pretesting step. Returning to the challenge setting, the performance measures chosen to evaluate the different iris/non-iris segmentation strategies were a set of quite classical ones used for binary classification: accuracy, precision, sensitivity, specificity, Pratt, F1 score, Rand index, global consistency error, E1 score, Pearson correlation coefficient. Final recognition, when included in the participant methods, was carried out in verification mode (1:1 matching). The corresponding performance measures were decidability index, equal error rate (EER), and receiver operating characteristic (ROC) curves with corresponding area under curve (AUC). Since MICHE-I was especially focused on iris segmentation, more metrics were used to measure performance in this operation. Moreover, when present, the results of proposals also addressing iris recognition were analyzed concentrating on the segmentation methods allowing a more reliable feature extraction and matching thanks to a better separation of the eye regions.

4.4.1 Metrics Used to Evaluate the Segmentation Quality

Table 4.2 summarizes the performance measures used to evaluate the candidate methods in MICHE-I, and that are quite common for classification problems (in our case, iris/non-iris) or even multi-class problems. Some of them are specifically suited for segmentation: Pratt metric is introduced in [49] and global consistency error in [44].

Each of the metrics exploited to evaluate the segmentation quality is able to capture some specific aspect of a correct classification. Of course, it is firstly important to correctly classify an existing edge pixel (true positives vs. false negatives) and this ability is different from avoiding false positives (vs. true negatives). Actually, the two can be in contrast, so that the lower the rate of false negatives, the higher the number of false positives could be. This is common for binary classifiers, and as a matter of fact they do play an asymmetrical role in evaluating segmentation algorithms: An algorithm that achieves the former might be less effective to achieve the latter. Errors in either direction can differently affect the rest of the processing. A contour interruption caused by false negatives can hinder if not completely compromise the detection of a shape or produce an unconnected contour where a connected one is needed/expected. The role of the first four metrics is to measure these aspects separately. F1 score rather provides an overall estimate of the ability of the algorithm to distinguish true edge pixels from false ones without missing too many of them. RI is a measure of the overall agreement between positive/negative classifications and ground truth, taking into account pairs of corresponding pixels. It can be extended to more different candidate classifications. E1 score represents a kind of complementary measure, since it rather measures the proportion of disagreeing pixels. Pratt metric evaluates accuracy from a point of view more strictly related to the specific segmentation problem, since it returns a global estimate of the distance between the detected contours and the ground truth: not only true/false, not

Table 4.2 Performance measures used to evaluate the methods submitted to MICHE-I

Accuracy	Accuracy measures the proportion of true results, summing up true positives and true negatives and computing the rate with respect to the total number of samples
Precision	Precision measures the proportion of the true positives against all the returned positive results that include both true and false positives
Recall	Recall is also called the true positive rate, or the sensitivity, and measures the proportion of positives that are correctly identified as such, i.e., true positive against ground truth positives
Specificity	Specificity is the true negative rate and measures the proportion of negatives that are correctly identified as such, i.e., true negatives against ground truth negatives
F1 score	F1 score can be interpreted as a weighted average of the precision p and the recall r, and is defined as: $F1\text{-}score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$; it ranges from a best value of 1 and a worst value of 0
Rand index (RI)	The Rand index counts the fraction of pairs of corresponding pixels in given segmentation and ground truth, whose elements are either both labeled as edge or both labeled as non-edge, both in ground truth and in the returned segmentation
E1 score	E1 score represents the classification error rate of the algorithm on the input image and is given by the proportion of corresponding disagreeing pixels (that have a different label in the returned segmentation and in ground truth); it can be computed by the logical exclusive OR operator
Pratt	Pratt metric is defined as a function of the distance between correct and measured edge positions; it is also indirectly related to the false positive and false negative edges: $Pratt = \frac{1}{max\{E_G, E_D\}} \times \sum_{k=1}^{E_D} \frac{1}{1+\alpha+d_i^2}$ where E_G and E_D are the number of ground truth and detected edge points, respectively, d_i is the distance from the *ith* detected edge point and the closest ground truth one, and α is a scaling constant that in the original metric formulation is $\alpha = \frac{1}{9}$; this metric takes into account the global trend of the distances between returned and ground truth edges; it ranges between an optimal value of 1 and a minimum of 0
Global consistency error (GCE)	The global consistency error evaluates at which extent one segmentation can be viewed as a refinement of the other, given two segmentations S_1 and S_2, a pixel p_i and regions $R(S_1, p_i)$ and $R(S_2, p_i)$. Containing the pixel in segmentation S_1 and S_2, respectively, a local (asymmetric) error measure is defined as $E(S_1, S_2, p_i) = \frac{\lvert R(S_1,p_i) \setminus R(S_2,p_i) \rvert}{\lvert R(S_1,p_i) \rvert}$, so that it is possible to compute a local refinement error in each direction at each pixel; the global consistency error forces all local refinements to be in the same direction, being finally defined as $GCE = \frac{1}{n} min\{\sum_i E(S_1, S_2, p_i), \sum_i E(S_2, S_1, p_i)\}$ with n the number of pixels; substituting the minimum of the sums with the sum of the minima provides the local consistency error (LCE) that would rather allow refinement in different directions in different parts of the image
Pearson correlation coefficient (PCC)	The Pearson correlation coefficient is a measure of the linear correlation between two random variables X and Y, returning a value between +1 and 1 inclusive, where 1 is total positive correlation, 0 is no correlation, and -1 is total negative correlation.

only lack of correspondence, but also distance from the true result. In this respect, GCE provides a similar yet more "directed" result, taking into account the direction of the error too: It measures how the errors with respect to ground truth (the direction is fixed) can result in a less detailed segmentation though bringing much the same core information. Finally, PCC is the usual Pearson correlation, to evaluate if the result segmentation and the ground truth present a similar trend.

4.4.2 Methods Participating in MICHE-I

Since most proposals included both segmentation and recognition, both sub-proposals will be briefly summarized when appropriate. A first observation deriving from the analysis of the methods is that it is a common practice to try to compensate for poor image quality using different approaches. A frequent one entails the use of the periocular region as an extra source of information. When the resolution of the iris region is not sufficient, or too many distortions are present, recognition can be supported by additional features extracted from the region around the eye. Among the other proposals along this line, this approach had been proposed also in NICE II challenge (addressing recognition) and in particular by the winning method by Tan et al. [65]. A combined strategy using more sets of features/methods can reduce the specific sensitivity to any particular data covariate. Last but not least, it is possible to exploit color compensation techniques to attenuate the typical cross-device difference of sensor features. The participating methods are listed below by alphabetical order of the first author's last name.

The approach by Abate et al. [4] for the challenge relies on an algorithm based on the watershed transform for iris segmentation, namely watershed-based iris detection (BIRD). The first step is to compute for each RGB channel the gradients in a colored, illumination-corrected image. The final gradient image is obtained by averaging the gradients computed over the separate channels. The watershed transform exploits the topographical distance approach [60]. The output of the watershed transform guides the binarization of the original image and the circle detection step, in order to find a parametrized expression for both the pupil and the sclera boundaries. Also, this proposal exploits the periocular region, which is localized using as reference the length of the iris radius. Differently from Santos et al. (see below) that exploit a rectangular periocular region, BIRD relies on a different choice. Starting from the approximating circle detected during the iris segmentation process, BIRD exploits its center coordinates and radius to construct two concentric ellipses that enclose part of the area around the iris. Both ellipses are centered in the center of the iris, but they are defined by different major and minor axes, always determined starting from the iris radius. The area enclosed by the ellipses is processed in a way similar to the Daugman rubber sheet model. The resulting rectangular region has a resolution which depends on the granularity chosen for the parameters (angle) and on the mapping. This way of processing the periocular region is quite original. For both regions, feature encoding is done by means of 64-bit color histograms, matched using the cosine dissimilarity and Hamming distance. Iris and periocular results are fused at a score level, by implementing a simple sum approach.

The proposal by Barra et al. [18] also includes segmentation and recognition. The segmentation method, named IS_IS, was originally proposed in [23]. The original method entails using Canny edge detector to identify edges, and then the pupil boundary is identified by a voting scheme that ranks circular edges by uniformity of the inner region and contrast between the inner and outer regions. Iris is identified after applying the rubber sheet model to the image, by inspecting the dark-to-light intensity variations along the columns of the image in polar coordinates. The method is modified to run on mobile devices. Feature encoding relies on spatial histograms (spatiograms). Common histograms can be considered as first-order spatiograms, while higher-order ones contain further information relating to the spatial domain spanned by the pixels falling in each bin. Second-order spatiograms used here store also the mean and covariance matrix of the pixel coordinates. Spatiograms are matched by correlation-based techniques.

The proposal by Haindl and Krupička [33] focuses on the detection of the non-iris components, especially reflections, for the parametrizations of the iris ring. The accurate detection of eyelids and reflections can have a significant impact on the final iris segmentation. The proposed model adaptively learns its parameters on the iris texture part and then searches for iris reflections by the recursive prediction analysis. After detecting reflections, pupil parametrization is carried out by form fitting techniques. Next, data is converted into the polar domain according to the usual rubber sheet model technique. In the resulting stripe, a texture analysis phase determines the regions of the normalized data that should not belong to the iris, according to a Bayesian paradigm.

As other proposals, Hu et al. [34] apply a combined approach and fuse different iris segmentation techniques, selected according to their performance in addressing specific cases of degraded images. Their proposal implements a model selection strategy, which selects the final parametrizations for iris and pupil boundaries among the candidates returned by the used baseline segmentation strategies. The selection relies on the image description provided by histograms of local gradients that are inputted to a support vector machine providing the fused response. This strategy is designed to be modular and can be updated by adding/substituting baseline segmentation methods. This proposal does not entail either coding or classification.

The method submitted to the challenge by Santos et al. [61] entails both segmentation and recognition. It uses both the information from the iris and from the periocular region, encoded/matched in a localized way. The first step is iris ring segmentation, which is carried out according to a variation of the integro-differential operator by Daugman. As a matter of fact, the characterizing part of this proposal is the encoding/matching step. Once identified, this ring also allows locating the periocular region of interest (ROI). According to a combined strategy, information encoding exploits a family of texture descriptors used separately in the iris ring and in the regions surrounding the cornea (i.e., eyelids, eyelashes, skin, and eyebrows). In particular, the periocular region undergoes a twofold examination, entailing both a distribution-based analysis of patches defined over a fixed grid and a global analysis of the whole region. The former is carried out by computing both local binary patterns (LBPs), histogram of oriented gradients (HOGs), and uniform LBP (ULBP). Each descriptor is computed separately for each patch and quantized into histograms. Global analysis rather entails feature extraction from the whole periocular ROI. In this case, the descriptors applied are scale-invariant feature transform (SIFT) and GIST (a set of five descriptors originally introduced in [47] to model the shape of a scene in a way that bypasses the segmentation and the processing of individual objects or regions). Iris information is encoded according to the classical approach described by Daugman [22]. It may appear that much more information is captured and stored from the periocular region than from the iris. During matching, scores from all the adopted descriptors are fused by a nonlinear supervised neural network. It is worth pointing out that the method further exploits device-specific calibration techniques that compensate for a different color rendering characterizing each experimental setup. The latter is especially useful in cross-sensor tests. In summary, this method uses information from two different sources, the iris and the periocular region, and further uses different descriptors to fully exploit their different characteristics. The overall proposal is a good example of how a difficult setting can be addressed by an ensemble of techniques.

4.4.3 Some Interesting Notes on Achieved Results

We will not report the detailed results of the competition. The interested reader can refer to [24]. The same will be done for the results from MICHE-II. Rather, we will underline some interesting aspects as possible guidelines to take into account.

For each method and for each device, the segmentation was carried out on images captured both indoor and outdoor (OUT); in some cases, no segmentation at all was returned, and this represents a kind of failure to enroll (FTE) error. The method by Haindl and Krupička achieves the highest rate of successfully segmented images, while the method used by Barra et al. achieves the lowest. However, thanks to the number of different performance measures exploited, it was possible to observe that the usable segmentation results returned by the latter, although less in number, were more accurate, providing the highest level of similarity with the ground truth. From the point of view of similarity with ground truth, the second method achieving the best results was the one by Abate et al. Therefore, these two methods should provide higher-quality masks, since they apply a more strict quality criterion for the obtained segmentation. Surprisingly enough, the methods by Haindl Krupička and by Hu et al. were instead more reliable in terms of the rate of success in the following recognition step. This seems a contradiction; therefore, a more careful investigation was carried out entailing the comparison of the 50 best common segmentations from the different methods. When working on the pictures where the segmentation task is easier, the results are different from those above. The mean scores on all devices testify that the method by Haindl and Krupička is actually the most reliable one. This means that it is not possible to predict the behavior of any method when problematic samples are submitted.

It is interesting to have a look at Fig. 4.9 to appreciate the differences that can be observed in the segmentation masks, whose accuracy can be influenced by the capture condition (indoor vs. outdoor), by the resolution of the image, determined by the capture device, and by the segmentation method.

It is possible to observe the frequent degradation of image quality passing from indoor to outdoor conditions, caused by a huger presence of reflections. However, indoor capture can be influenced by the different color temperatures of the illumination sources and of the sensor (see, e.g., IP5 IN and GS4 IN in Fig. 4.9). A higher resolution can be desirable to capture finer details. However, also the amount of noise can be greater. An example is given by the more stable segmentation produced on GT2 images. Of course, a final assessment can only be provided by the recognition results obtained on images captured in corresponding conditions.

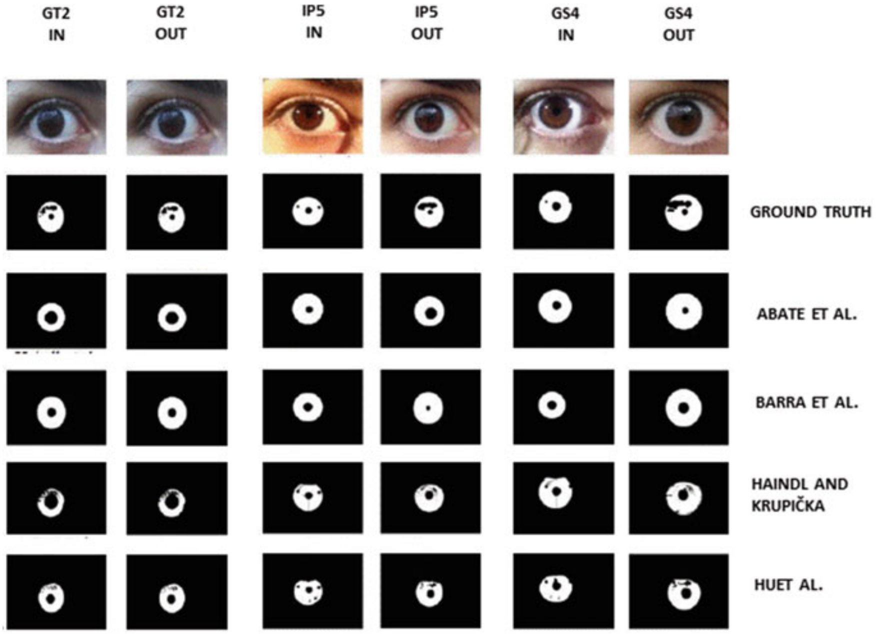


Fig. 4.9 Examples of the segmentation results in good-quality MICHE images

4.4.4 Recombination of Segmentation and Recognition Modules

Recognition was not specifically addressed in MICHE-I. However, since most submitted proposals also included a recognition module, this allowed to carry out further evaluations. We recombined segmentation and recognition modules in different ways. The aim was to evaluate how the two parts of a proposal depended on each other and which segmentation methods provided a more robust preliminary step for a different coding. The well-separable and not already published segmentation modules were those included in the proposals by Abate et al. [4], Barra et al. [18], Hu et al. [34], and Santos et al. [61] that have been described above.

The performances achieved by the recognition methods included in the proposals submitted to MICHE-I were evaluated in verification mode (1:1 matching, where it is to intend that a probe subject is matched against a single gallery subject, though possibly exploiting more templates per subject). The used figures of merit (FOMs) were decidability, area under curve (AUC) with reference to the receiver operating characteristic (ROC) curve, and equal error rate (EER). The preliminary identification of the iris ROI was carried out in turn using the segmentation methods of the challenge proposals. Decidability is the same FOM used for the NICE II competition [51]. The first step to compute it requires to carry out a “one-against-all” comparison for each image $I = I_1; \ldots; I_n$ of the dataset. The matching process exploits the segmented images and the corresponding binary maps $M = M_1; \ldots; M_n$ that provide the noise-free iris region identified by the segmentation step. The comparison provides a set of intra-class dissimilarity values $DI = DI_1; \ldots; DI_k$, with k the number of image pairs belonging to a same iris, and a set of inter-class dissimilarity values $DE = DE_1; \ldots; DE_m$, with m the number of image pairs belonging to different irises. The decidability value $d'(DI_1; \ldots; DI_k; DE_1; \ldots; DE_m) \rightarrow [0; \infty[$ used as evaluation measure is computed separately for each recognition method as:

$$d' = \frac{|avg(DI) - avg(DE)|}{\sqrt{\frac{1}{2} \times (\sigma^2(DI) + \sigma^2(DE))}}, \tag{4.1}$$

where avg and σ^2 have the conventional meaning of average and variance functions computer over the parameter sets.

The challenge proposals including a clearly separable recognition module were those by Abate et al. [4], Barra et al. [18], and Santos et al. [61]. A further recognition method was submitted for a special issue based on but not limited to the challenge, namely the one presented by Raja et al. in [55]. It was tested in combination with all the four segmentation modules in order to get a wider set of experiments. This approach to feature extraction and recognition is based on deep sparse filtering. Sparse filtering [45] is an unsupervised algorithm which does not explicitly aim to model the distribution of data. It optimizes a simple cost function of sparsity using l_2 normalized features. The only parameter required in learning sparse filters is the number of features, as the sparse filters are learned by optimizing sparsity in feature distribution. Extending the method to deep sparse filters, a variable number of layers form the building blocks in learning. In the proposed approach, the deep sparse filter consists of two layers such that layer 1 is trained using 200,000 random patches of size 16 × 16 pixels from 4212 natural images. The sparse filtered features obtained as output from the layer 1 are normalized and submitted to layer 2 using a feedforward network. The sparse filter is trained at layer 1 with 256 filters of dimension 16 × 16 features and at layer 2 with 256 sparse filters of 16 × 16 features. The sparse filter features obtained from layer 2 are exploited to extract features from the iris images. Each iris image is convolved with the 256 filters of layer 2, so that a total of 256 response images are obtained. These images are binarized and pooled at pixel feature level in groups of eight so as to obtain a response image from each pool. Afterward, an histogram is extracted from each obtained image and histograms are chained to form a feature vector.

When recognition is assessed in a context like the one addressed by MICHE, it is significant to also test cross-device performance. Segmentation can affect the final process by a different accuracy in identifying pixel regions belonging to the iris. Some such regions may be missing, or some non-iris patches may erroneously enter the feature extraction and matching step. The first kind of errors becomes critical if quite extended, so that relevant information may be left out from the matching. This may either cause a FA (the missing region was a highly characterizing one) or a FR. The second hypothesis is less frequent, because the remaining part of the iris could be sufficient for a positive recognition. On the other hand, the second type of error may affect the final result even if less extended, since it introduces differences between two irises even where pixels should not have been considered for the matching, therefore erroneously increasing the differences. The most frequent consequence is a FR, since in general the non-iris regions have a structure significantly different from the iris regions of both the same eye and different eyes. Differently from segmentation, feature extraction and matching rely on finer details that allow to summarize the microstructure of the iris region. In this case, artifacts and sensor typical noise introduced by a sensor can cause a higher accuracy degradation when captured images are matched against those captured by a different sensor. In particular, the sensor pattern noise (SPN) as defined in [42] is so specific of each device, though of the same brand and model of others, that it could be used to identify the one that captured an image.

Cross-sensor matching experiments were implemented by alternatively using sets of images acquired by the same device as either gallery set or test set (probe set), including intra-device recognition. Each combination of probe–gallery devices will be referred to as a *class of comparison*. Detailed report of experimental results can be found in [24]. It is interesting here to just underline the main observed aspects.

The first observation deriving from the inspection of the results is that the recognition method by Santos et al. systematically outperforms the others in all classes of comparisons and with all segmentation algorithms. Among the classes of comparison, those entailing the same device generally allow better performance with respect to heterogeneous pairs. The class of comparison GT2vsGT2 achieves a higher level of performance (on average and compared to the others) in terms of both the EER values achieved by the various combinations of segmentation/recognition, and of the relationship between FARs and FRRs (better ROC curves). This happens notwithstanding the poorer resolution of the embedded camera. However, this class generally presents the lowest decidability values. The apparent contradiction may be caused by the fact that the sizes of probe and gallery sets for GT2 are smaller than the others. This means a lower percentage of inter-classes uncertainty, which contributes to increase the level of performance and to a lower intra-class generalizability of the results. Another non-obvious observation is that performances are sensitive to the swap of the probe/gallery role of images from different sources. Of course, this depends on the different levels of detail of images from devices with a different resolution. In general, the higher the resolution of the probe (the amount of details) with respect to the gallery images, the worse the result, because part of the probe information does not get matched. Finally, the segmentation methods by Haindl and Haindl and Krupička , and by Hu et al. provide more stable results that cause less performance difference in the following recognition step, even if in both cases the superiority of the recognition by Santos et al. is even more evident.

A final analysis of the results of MICHE-I when using a single recognition system was carried out to identify the "intrinsic" covariates that can mainly affect recognition. For this reason, "extrinsic" covariates were neglected. In particular, for each experiment the device was fixed for both probe and gallery, to neglect factors related to the device difference. Moreover, also the segmentation and the recognition methods were fixed, in order to neglect the differences in the achieved similarity given by the specific techniques. The aim was to identify the best/worst pairwise comparisons that were common to all experiments. For each experiment, each gallery template is compared with each of the others of the same subject, and the full set of the obtained intra-subject dissimilarity scores is organized in a list ordered by ascending values. Given device and method peculiarities, such scores may fall in different ranges and have different distributions across the experiments. However, it is still worth comparing the obtained rankings. The samples considered as the "best" ones always appear on the heading part of the ordered lists, meaning that they always achieve a very good similarity when compared with samples of the same subject. The contrary holds for "worst" samples. The possible recurrent features of the latter are the most interesting, because they represent those intrinsic conditions that can hinder a correct recognition. Figure 4.10 shows some typical examples of "worst" samples. It is possible to observe that the occlusions by the eyelids are rather evident in most of the pictures; the average brightness of images is low, or the iris falls in a shadow region; reflections of unpredictable nature can affect images captured outdoors (image in the upper left corner of Fig. 4.10). On the contrary, in "good" samples the visibility of the irises and of the pupils is high, thus making it easier to detect and segment them.

Aiming at a possible gain in accuracy, the combination of multiple recognition methods was evaluated. In this last round of experiments, the different recognition results were fused at score level. Each experimental session was identified by the pair of (possibly different) devices capturing gallery and probe, by the segmentation method, and by the recognition strategy exploited. The latter can entail either a single

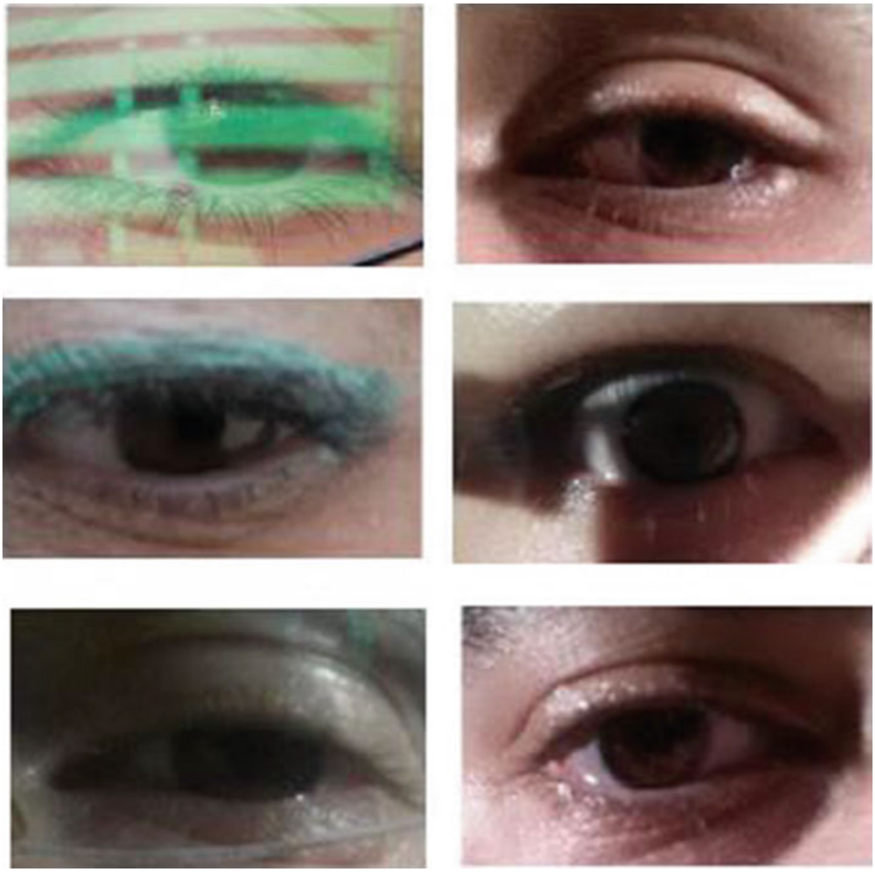


Fig. 4.10 Examples of "worst" samples

method or a score-level fusion of the results from a possible subset of them. For each session, each recognizer involved produced a dissimilarity matrix: The lower the value of a cell (score), the higher the probability that the images on the row and on the column depicted two irises from the same subject. In order to fuse more results, the values in the matrices had to be normalized, in order to obtain comparable values in a common range [0, 1]. This was achieved by the min/max rule, by considering the minimum and the maximum value for each matrix. Two score-level fusion strategies were investigated: the *simple sum* fusion and the *matcher weighting* fusion. The former consists in just summing up the scores produced by each of the M methods involved in a session. The values in the obtained distance matrix are normalized again to remain in the range [0, 1]. The *matcher weighting* fusion assigns to each matcher m a weight w_m that is inversely proportional to the achieved EER e_m and is defined as follows:

$$w_m = \frac{\frac{1}{\sum_{m=1}^{M} \frac{1}{e_m}}}{e_m}, \tag{4.2}$$

where $0 \leq w_m \leq 1$ and $\sum_{m=1}^{M} w_m = 1$.

The number of segmentation methods, of recognition methods, of pairs of probe/gallery devices, and the consequent number of their combinations, makes the amount of results to analyze and report extremely large. It is easy to guess how this amount further increases by introducing possible combinations of recognition methods in multimodal strategies and possible different fusion strategies for each such combination. Once again, we report here only the most relevant outcomes. Given the four segmentation methods, and for each of them the nine combinations of probe/gallery devices, an overall analysis of the fusion results testifies that the improvement achieved by using any of the two fusion strategies is rather limited. In many cases, the AUCs are just a little wider than the ones obtained by an execution of Santos et al. algorithm alone. This means that the four recognition methods taken into account have no sufficiently complementary ability to extract and match relevant features. In other words, they rely on similar information content, though represented in different ways. The increased computational demand required by running different methods and by the fusion of their results is not positively counterbalanced by a significant enough improvement in the recognition accuracy. In conclusion, a generally well-performing method can achieve better performance than the combination of weaker ones if the latter do not represent different kinds of information so as to balance each other's flaws.

4.5 MICHE-II Challenge: Iris Recognition

Along the line of NICE challenges, the second round of MICHE challenge, namely MICHE-II, focused on iris recognition. As already underlined, the accuracy of the encoding in correctly extracting relevant and discriminative features, and the following recognition, can be generally heavily affected by the quality of the segmentation. In order to provide a common starting point to all participants, not only a common benchmark was provided that represents an extension of the previous dataset though maintaining the same feature distribution and variety. As for the second phase, all the competing methods had to start from the results of the same segmentation algorithm, in order to be able to assess the net contribution on the feature extraction/recognition alone. As for NICE, the best segmentation algorithm from MICHE-I was chosen, namely the one by Haindl and Krupička [33]. Of course, different feature extraction procedures can produce different templates and specific approaches to similarity/distance evaluation. Therefore, the competitors were free to choose a suitable distance measure for the produced iris templates, with the only constraint to be a semi-metric. The higher the dissimilarity, the higher is the probability that the two irises are from different subjects. Given I the set of images from the MICHE database, and I_a and $I_b \in I$, the dissimilarity function D had to be defined as:

$$D : I_a \times I_b \rightarrow [0; 1] \subset R, \tag{4.3}$$

with properties

$$D(I_a; I_a) = 0 \tag{4.4}$$

$$D(I_a; I_b) = 0 iif I_a = Ib \tag{4.5}$$

$$D(I_a; I_b) = D(I_b; I_a) \tag{4.6}$$

Each algorithm had to return a full dissimilarity matrix among probe and gallery sets. New images were added, and distance matrices were computed from scratch during the evaluation of the methods, in order to avoid any kind of bias while creating the final rank. Distance matrices were used to compute the classical FOMs to rank them, namely recognition rate (RR) for identification and receiver operating characteristic (ROC) curves, in particular the area under curve (AUC), for verification.

4.5.1 Methods Participating in MICHE-II

We summarize below the main characteristics of the participants. As for MICHE-I, the methods are listed by alphabetical order of the first author's last name, even if they were assigned an identifying label. Also in this case, a special issue following the challenge hosted a further method that was evaluated and compared in a second round.

The method labeled as *irisom* was implemented is first described in [1], and experiments are extended in [3]. It implements iris recognition in the visible spectrum through unsupervised learning by means of self-organizing maps (SOMs). The proposed method starts with a first step of image enhancement by simple image processing techniques, like contrast enhancement and histogram adjustment. Then, it exploits unsupervised learning by self-organizing maps (SOMs). The SOM network clusters iris features at pixel level, after discarding those marked as non-iris in the segmentation mask. The discriminative feature map is obtained by using RGB data of the iris combined with the statistical descriptors of kurtosis and skewness, computed at pixel level in a neighborhood window of size 3 × 3. The network produces a feature map with the activation status of the neurons for each pixel. The map represents a cluster decomposition of the image, which maps the problem of iris recognition onto a lower-dimensional space. The method then computes the histogram of gradients (HOGs) over the obtained feature maps, and the result is used as a feature vector. Verification relies on the Pearson correlation coefficient computed in the [0,1] real interval. The best results for this method were achieved with 5 × 5 and 10 × 10 SOMs.

The method *otsedom* is described in [7], and experiments are extended in [5]. The proposal was submitted by a joint team from Universidad del Pais Vasco (UPV) and Universidad de Las Palmas de Gran Canaria (ULPGC). The approach combines popular computer vision techniques and machine learning paradigms. Starting from the segmentation, well-known local descriptors are computed. Those suitable for the problem are selected after evaluating a collection of 15, with different grid configuration setups. Popular examples of the selected descriptors are local binary patterns (LBPs), local phase quantization (LPQ), and Weber local descriptor (WLD). They are used individually to build separate classifiers by a supervised machine learning approach. Each classifier computes the dissimilarity between two irises by the histogram distance between the two a-posteriori probability distributions computed from the two iris images. In a second step, the best combination of subsets of classifiers is evaluated to build the best multi-classifier system out of the individual ones. In practice, the final algorithm combines the best five descriptors to obtain a robust dissimilarity measure of two given iris images. The mode of each a-posteriori probability for each class value is used to combine the classifiers. Some combinations of local descriptors also take into account the periocular region.

A research group with a slightly different composition from the above presented a further set of proposals collectively labeled as *ccpsiarb* [8]. The experiments presented by the authors are extended in [6]. Based on the training dataset given by MICHE-II, a set of classifiers is constructed and tested, aiming at classifying a single image. Iris images are processed using well-known image processing algorithms. Different transformations of the original pictures can highlight different characteristics of the images. Examples of the transformations tested are equalization, Gaussian, median, etc. This phase aims at expressing the variability in the aspect of a picture, so as to obtain different values for the same pixel (feature) positions. The output images are considered the input of the previously trained classifiers, obtaining the a-posteriori probability for each of the considered class values. The classifiers implement some well-known ML supervised classification algorithms, with completely different approaches to learning: IB1, Naive Bayes, random forest, and C4.5. Experiments take into account the 19 image collections obtained by applying single transformations, and the four different classifiers, giving a total of 76 experiments. After testing all these combinations, the edge transformation followed by IB1 classification (identified as combination *ccpsiarb_17*) is identified as the combination providing the best results.

The *tiger_miche* method described in [9], with experiments extended in [10], uses a combination of a popular iris code approach and a periocular biometric based on the multi-block transitional local binary patterns. To generate iris codes, the method convolves the unwrapped iris image with 1-D Log-Gabor filter. Log-Gabor functions are chosen because they have no DC component, and this can alleviate the negative influence of light intensity differences on textural information, which affects the images captured in the visible spectrum. Since a 1-D filter is used, each row in the unwrapped image is treated as 1-D signal. It is multiplied in a frequency domain with 1-D Log-Gabor filters of different scales that capture textural information with different levels of details. To generate the iris code, the phase information of the output signals is quantized into four levels, one for each possible quadrant in the complex plane. The coding then discards iris code values at positions corresponding to either very small or very large amplitudes of filter response. Hamming distance is used to match the iris codes, once adapted to take the segmentation mask into account.

Transitional LBP (TLBP) uses comparisons between neighbor pixels in a clockwise direction for all pixels, except the central one, so that it encodes information about the partial ordering of border pixels. Its formulation is

$$TLBP_{P,R} = s(g_0, g_{P-1}) + \sum_{i=1}^{P-1} s(g_i - g_{i-1})2^i, \tag{4.7}$$

where, as usual, g_0 is the central pixel of the window over which the code is computed, P is the number of neighbors, R is the window radius, and $s(x)$ returns 1 or 0 according to the sign of its argument. Multi-block extensions of both LBP and TLBP use the average gray values from the blocks of pixels instead of the gray values of individual pixels to create the code. The method uses block sizes 3×3, 9×9, and 15×15. For each block size, it computes $TLBP_{12,3}$ and $TLBP_{24,6}$ codes and their histograms, which are concatenated to create a feature vector. Histogram vectors are matched using chi-square distance between the concatenated histograms.

The Hamming distance between two iris codes and the periocular matching score are computed separately and then combined by a score-level fusion to improve the system accuracy. The values returned by the matchers fall in different ranges and present very different score distributions; therefore, the authors exploit z-score normalization.

The method labeled as *karanahujax* is described in [11], with experiments extended in [12]. It exploits a hybrid convolution-based model, for verifying a pair of periocular images containing the iris. The baseline proposed model is based on root scale-invariant feature transform (SIFT). The binary mask is used to get the iris image rid of occlusions. Then, dense color root SIFT descriptors [15] are computed, giving keypoints with identical size and orientation. The hybrid model is conceived as a combination of this baseline model and of two deep networks, an unsupervised one and a supervised one. The unsupervised convolution-based deep learning approach (Model1) uses a stacked convolutional architecture, with external models learned a-priori on external facial and periocular data, on top of the baseline root SIFT model: The approach is completed by different score fusion strategies. The supervised approach (Model2) also uses a stacked convolution architecture, but the feature vector is learned in a supervised manner. The fusion carried out in the hybrid model exploits an average of the computed scores after suitable normalization.

FICO_matcher exploits the fast iris recognition (FIRE) algorithm described in [29]. Related experiments are extended in [30]. The key features of the method are the use of a combination of classifiers exploiting the iris color and texture information, and its limited computational time that makes it particularly attractive for fast identity checking on mobile devices. The classifiers whose results are fused, respectively, exploit the distance among color, texture, and "cluster" features, meaning the presence of specific pixel aggregations in the image. In order to compute color distance, given two irises, each picture is first split into small blocks. For each pair of corresponding blocks, the color distance is computed, and the minimum color distance obtained is the final score returned by the color descriptor. The exploited color distance measure is the Kolmogorov–Smirnov distance. Given the cumulative histograms of images, with $\hat{h}_i = \sum_{j\leq i} h_j$, the distance is defined as:

$$d_{K-S}(H,K) = \max_i(|\hat{h}_i - \hat{k}_i|). \tag{4.8}$$

The texture descriptor relies on the Minkowski–Bouligand dimension (box-counting dimension). The box-counting dimension of a set S is defined as:

$$dim_{box}(S) = lim_{\epsilon\rightarrow 0}\frac{logN(\epsilon)}{log\frac{1}{\epsilon}}, \tag{4.9}$$

where $N(\epsilon)$ is the number of boxes of side length ϵ required to cover the set S. Images are decomposed in layers according to the colors, each layer is divided into blocks, and for each of them the box-counting dimension is computed. These are finally chained in a feature vector, and matching relies on Euclidean distance.

"Clusters" are connected components resulting from morphological operators applied to image layers obtained as for texture description. The features characterizing clusters are centroid coordinates, orientation, and eccentricity. Such cluster feature vectors are chained to make up an image feature vector.

Two versions of the method are tested, namely V1 that uses three kinds of descriptors, by suitably weighing the obtained distances, and V2 that does not use texture.

The method *Raja* described in [56] did not participate in the first round of the MICHE-II competition, but was submitted for the following special issue and was therefore tested from scratch together with the others. It proposes deep sparse filtering carried out on both multiple image patches and on the complete image. The image corresponding to each RGB channel is divided into a number of blocks. Both such blocks and the whole image are processed to obtain deep sparse histograms using the set of deep sparse filters. The final feature vectors are the concatenation of the set of histograms obtained from different channels and blocks. The extracted features are represented in a collaborative subspace, to jointly represent the set of training samples that correspond to enrollment. In such space, a new classification approach is adopted.

4.5.2 Some Interesting Notes on Achieved Results

The ranking of the participant methods was obtained by running all the methods from scratch at BipLab—University of Salerno, over an extended set of images after segmenting them with the segmentation algorithm provided for the competition. The final rank list in Table 1 reports the best performing version among the ones submitted for each author (label). The rank was obtained by averaging the recognition rate (RR) and the area under curve (AUC) achieved, and considering only images captured by the two smartphones. Both cross-device (ALLvsALL) and single-device settings were considered

- *tiger_miche*
- *karanahujax_Model2*
- *Raja*
- *irisom_10x10*
- *FICO_matcher_V1*
- *otsedom*
- *ccpsiarb_17.*

As for the segmentation results, details can be found in [25], while it is interesting here to point out some interesting aspects of the outcomes.

As a first observation, the better the ranking achieved, the more stable the method with respect to the test setting. Of course, the hardest conditions are those found in ALLvsALL, where gallery and probe images come from different devices in unpredictable pairings. As expected, all methods provided consistently lower performances in this condition. The results confirmed the observation stemming from MICHE-I outcomes: The images over which the highest recognition accuracy was achieved in homogeneous settings (gallery and probes from the same device) come from IP5, and the achieved scores further present a lower standard deviation, notwithstanding the lower resolution of the camera. Once more, this seems to suggest that, in the given uncontrolled and noisy conditions, higher resolution may also increase the way the noise typical of iris images can affect recognition. A related observation regards the way the different methods behave with respect to the different devices. The best method achieves high results with both cameras. Four methods, namely *karanahujax_Model2, irisom_10x10, FICO_matcher_V1*, and *ccpsiarb_17*, rather achieve their best performance with IP5. Methods developed in more versions are more stable w.r.t. the different variations. For instance, *karanahujax_Model1* and *karanahujax_Model2* achieve the same final score, but *Model2* achieves the better behavior in ALLvsALL. A similar constant behavior is observed for the many versions of *ccpsiarb*, while *FICO_matcher_V2* achieves dramatically worse results than *FICO_matcher_V1*, confirming the expected outcome that texture distance is critical for iris matching. Execution times were not evaluated in the competition, but are important for real-time operations. The best method *tiger_miche* also achieved the best result in terms of time required by the single matching operation. Only

FICO_matcher_V1 did better, and even more *FICO_matcher_V2* that, however, provides much lower recognition accuracy. On the other extreme, we find the methods relying on ML techniques, which therefore seem not suited for a real-time operational setting.

4.6 MICHE After the Challenges

The previous sections have shown the role of the MICHE dataset within the challenges using it as benchmark: Robust approaches have been designed, developed, and tested for both segmentation and recognition/verification purposes, mainly thanks to the dual nature of the dataset itself:

- The different acquisition modalities adopted for the enrollment of the subject, from the indoor/outdoor acquisition to the different illumination conditions, and most of all the different devices, have allowed the design and testing of cross-sensor verification algorithms.
- From a different point of view, the capture protocol assured a well-balanced presence of images presenting all the possible distortions that can affect iris images in realistic mobile unattended conditions.

In the last years, pattern recognition performance in terms of both accuracy and computing time has been considerably improved, mainly due to the wide diffusion of the artificial intelligence-based approaches like fuzzy controller configurations and machine/deep learning techniques. Therefore, it has become possible to address more complex problems and conditions, also in biometrics. As a consequence, despite the high level of complexity of the images in the MICHE dataset, and thanks to its characteristics, a number of researchers investigating iris recognition have used it also outside the challenge to train and/or test their architectures over these images, also reaching quite interesting results. These approaches have successfully addressed each of the issues involved in a typical iris recognition systems, as summarized in Fig. 4.11.

Eye Landmark Detection. The proposal in [35] deals with a novel approach for eye landmark detection with two-level cascaded convolutional neural networks. The network at the first level utilizes eye state estimation as an auxiliary task to provide the initial positions of the eyes. The shallower network at the second level fine-tunes eye positions by taking as input some small regions centered at predicted eye point locations.

Noise Removal. The goal of the work presented in [2] is to implement an effective lightweight fuzzy-based solution for noise removal from iris images, which allows a fast yet reliable segmentation approach which preserves the original resolution of the iris images.

Sclera Segmentation. Sclera segmentation can represent a preliminary step for either a correct iris identification or further processing based on the segmented area. The paper [13] proposes a new sclera quality measure and a method for sclera segmentation under relaxed imaging constraints. In particular, the quality measure is based on a focus measure. The sclera segmentation is obtained by fusing the information about pixel properties of both the sclera area and the skin around the eye. The authors also propose a template rotation for sclera alignment and distance scaling methods to minimize the error rates when noisy eye images are captured at-a-distance and on-the-move, together with overcoming head pose rotation.

Iris Segmentation. The method in [54] accurately localizes the iris by a model relying on the histograms of oriented gradients (HOGs) descriptor and on a support vector machine (SVM) classifier, namely HOG-SVM. Based on the achieved localization, the iris texture is automatically extracted by means of a cellular automaton which evolves via the GrowCut technique.

The study in [16] proposes a two-stage iris segmentation scheme based on a convolutional neural network (CNN), which is capable of accurate iris segmentation in severely noisy environments for iris recognition by visible light camera sensor.

The same group proposes in [17] a densely connected fully convolutional network (IrisDenseNet), able to determine the true iris boundary even with low-quality images. The approach ensures an improved information flow between the network layers, by introducing dense connectivity, i.e., the direct connections from any layer to all subsequent layers in a dense block. The experiments are carried out on five datasets, acquired in both visible and NIR light, including MICHE.

The segmentation method proposed in [14] is designed for the unconstrained environment of the smartphone videos. It is based on the preliminary choice of the best frames from the videos. Then, it tries to enhance the contrast of these frames between dark and light regions by applying two fuzzy logic membership functions on the negative image.

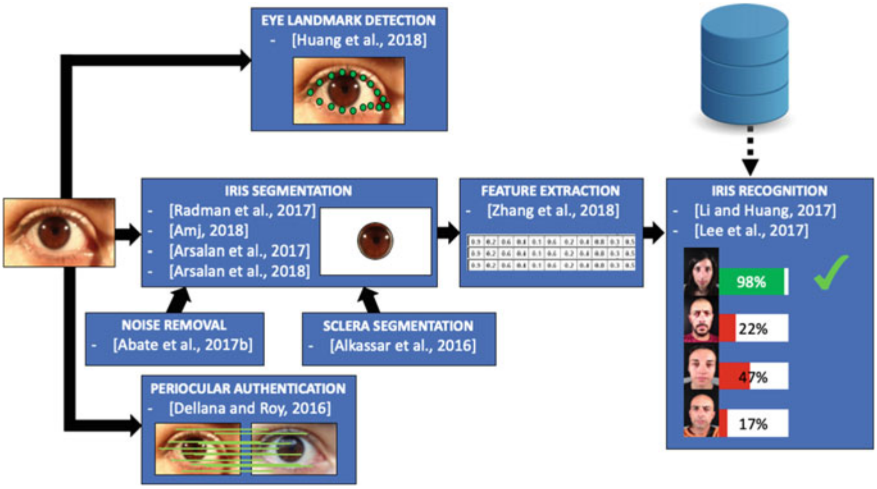


Fig. 4.11 A schema organizing the works that have used MICHE dataset according to the specific goal of the research

Feature Extraction. The proposal in [67] deals with a nonlinear dynamic data analysis tool, global preserving kernel slow feature analysis (GKSFA). This tool is able to extract the high nonlinearity and inherently time-varying dynamics of batch process, but, being an unsupervised feature extraction method, it lacks the ability to utilize batch process class label information. The authors propose a novel batch process monitoring method based on the modified GKSFA, namely discriminant global preserving kernel slow feature analysis (DGKSFA), which integrates discriminant analysis and GKSFA. MICHE dataset is used to exemplify discriminant and cluster analysis, to help explaining the proposed nonlinear contribution plot.

Iris Recognition. The paper [40] proposes an iris recognition mechanism to solve the problem of user authentication in wearable smart glasses. Given the premises, the contribution deals with both hardware and software. As for the hardware, a set of internal infrared camera modules is designed, including an infrared light source and a lens module, which is able to take clear iris images within 25 cm. As for the software, the devised iris segmentation algorithm is devised to be used on smart glass devices. Regarding the iris recognition, the authors propose an intelligent Hamming distance (HD) threshold adaptation method which dynamically fine-tunes the HD threshold used for verification according to empirical data collected. The research in [39] proposes a new recognition method for noisy iris and ocular images by using one iris and two periocular regions, both centered in the pupil and with a slightly different radius. The approach exploits three convolutional neural networks (CNNs).

Periocular Authentication. The experiments in [27] apply a convolutional neural network (CNN) to carry out periocular authentication on two datasets. Several different data augmentation techniques are tried to increase accuracy, and the results testify their relative benefits.

Miscellanea. MICHE has bee exploited as benchmark even for experimenting algorithms out of the scope of the “hard” biometric recognition (individual subject recognition), like in [58], in which a feasibility study of gender recognition from ocular images has been proposed. In an even wider scope, given the multiple cameras involved in the acquisition process of MICHE dataset, some works have used it to assess sensor identification methods. The purpose in that case is to classify the images according to the sensor that shot it. As an example, [28, 38] propose two approaches with this goal. They are, respectively, based on deep learning networks and on a technique based on photo-response non-uniformity noise (PRNU). Sensor features have also been occasionally exploited for binding the identity of a subject to the information related to the sensor of his/her smartphone, as shown in Fig. 4.12, in order to obtain a double check of the user fusing biometrics and hardware metrics. Interesting detailed analyses are reported in [31, 32].

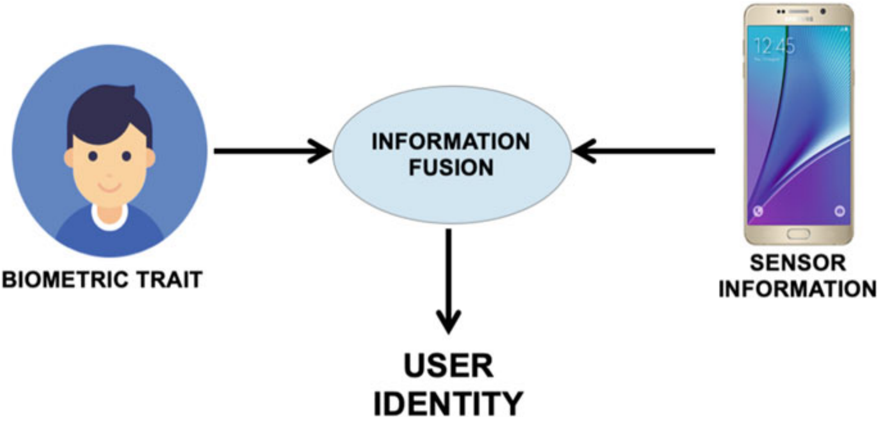


Fig. 4.12 Fusion of the information related to the subjects and those related to the smartphone sensor could both improve the verification of user identity and confirm the ownership of the device

4.7 Conclusions

This chapter addressed the challenges and difficulties in performing reliable biometric recognition, using self-acquired images from the subjects (*selfies*). In particular, we described the MICHE dataset, used as main data source for two international competitions about segmentation/recognition effectiveness of biometrics systems with such type of data. Based on our MICHE experience, it is possible to identify a number of take-home messages, presented below in the form of a list:

- An experienced operator could control specific critical conditions (e.g., pose, illumination, eye framing), possibly repeating the sample capture. However, this is not possible in uncontrolled/unattended conditions.
- The acquisition of the iris using visible light and in uncontrolled conditions presents peculiar difficulties, but it may rise even more problems when the operational setting entails a mobile application: It is necessary to compensate for users' lack of technical experience/ability and poor image quality, and also consider the possibly different features of the devices used for enrollment/recognition.
- Indoor conditions usually arise less illumination distortions with respect to outdoor, where a higher number of illumination sources may affect the image quality. On the other hand, data yielding from different indoor environments has high probability of being heterogenous, due to the different color temperatures of illumination sources.
- Reflections are more evident in outdoor than in indoor environments, but a more diffused and uniform illumination can create better conditions for localization and segmentation.

- Higher resolutions increase the amount of information collected, but also increase the levels of noise. Hence, the signal-to-noise ratio appears to be weakly correlated with the resolution of the images acquired.
- It is not possible to fully and reliably predict the behavior of any method when problematic samples are submitted; i.e., there is substantial amount of work to be done in terms of reliability and robustness of recognition in case of severely degraded samples.
- The periocular region, coded either by the same or by different descriptors than those used for the iris, can improve the recognition accuracy by providing additional information; as a matter of fact, using this multi-trait strategy has become a quite used solution, especially when expecting poor-quality eye samples.
- The combined use of multiple types of features can reduce the negative effect of a particular data covariate. However, at the same time, it tends to augment the computational complexity of the recognition chain, which might be particularly problematic for the execution in mobile devices.
- Typical cross-sensor differences may modify the iris micro-texture and possibly introduce artifacts, but several problems can be addressed by suitable color compensation techniques.
- It appears that the higher the resolution of the probe with respect to the gallery images, the higher the amount of unmatched information and, therefore, the lower the recognition accuracy. This observation suggests that a gallery update should be carried out when the sensor technology improves too dramatically.
- The increase of noise due to higher resolution might be limited to uncontrolled conditions where no capture adjustment is attempted, as in the case of the MICHE dataset; due to the lack of extensive cross-resolution tests in either controlled or uncontrolled conditions, it is not possible to generalize this observation.
- Intrinsic factors affecting the recognition problem (not related to either the capture device or the segmentation/recognition methods) are the iris occlusions due to eyelids, the low brightness of the samples, the existence of shadows in the iris region, and reflections of unpredictable shape and color inside the iris ring.
- Fusion of different features and/or different classifiers can improve the matching phase, when each component highlights and takes into account a different relevant aspect for coding and matching. However:
  - fusing more recognition methods can be effective only if they take into account sufficiently complementary information; this may not be true notwithstanding the different ways of representing features if the information content is basically the same;
  - it is not sufficient to fuse different computer vision techniques to enhance the image and different descriptors to capture different properties; it is also necessary to identify those processing steps able to extract the really relevant information.
- machine learning-based techniques seem still too demanding, especially in terms of the computational time cost, to be exploited in real-time operations in mobile devices, where the computing power is limited and the requirement of low energy consumption is a strong constraint.

References

1. Abate A, Barra S, Gallo L, Narducci F (2017a) Skipsom: Skewness & kurtosis of iris pixels in self organizing maps for iris recognition on mobile devices. Institute of Electrical and Electronics Engineers Inc., pp 155–159
2. Abate AF, Barra S, Fenu G, Nappi M, Narducci F (2017b) A lightweight mamdani fuzzy controller for noise removal on iris images. In: Battiato S, Gallo G, Schettini R, Stanco F (eds) Image analysis and processing—ICIAP 2017. Springer, Cham, pp 93–103
3. Abate AF, Barra S, Gallo L, Narducci F (2017c) Kurtosis and skewness at pixel level as input for SOM networks to iris recognition on mobile devices. Pattern Recogn Lett 91:37–43. Mobile Iris CHallenge Evaluation (MICHE-II)
4. Abate AF, Frucci M, Galdi C, Riccio D (2015) Bird: watershed based iris detection for mobile devices. Pattern Recogn Lett 57:43–51
5. Aginako N, Castrill-Santana M, Lorenzo-Navarro J, Martnez-Otzeta JM, Sierra B (2017a). Periocular and iris local descriptors for identity verification in mobile applications. Pattern Recogn Lett 91:52–59. Mobile Iris CHallenge Evaluation (MICHE-II)
6. Aginako N, Echegaray G, Martnez-Otzeta J, Rodrguez I, Lazkano E Sierra B (2017b) Iris matching by means of machine learning paradigms: a new approach to dissimilarity computation. Pattern Recogn Lett 91:60–64. Mobile Iris CHallenge Evaluation (MICHE-II)
7. Aginako N, Martínez-Otzerta J, Sierra B, Castrillón-Santana M, Lorenzo-Navarro J (2016a) Local descriptors fusion for mobile iris verification. In *2016 23rd International conference on pattern recognition (ICPR)*. IEEE, pp 165–169
8. Aginako N, Martínez-Otzeta JM, Rodriguez I, Lazkano E, Sierra B (2016b) Machine learning approach to dissimilarity computation: Iris matching. In: 2016 23rd International conference on pattern recognition (ICPR). IEEE, pp 170–175
9. Ahmed NU, Cvetkovic S, Siddiqi EH, Nikiforov A, Nikiforov I (2016) Using fusion of iris code and periocular biometric for matching visible spectrum iris images captured by smart phone cameras. In: 2016 23rd International conference on pattern recognition (ICPR). IEEE, pp 176–180
10. Ahmed NU, Cvetkovic S, Siddiqi EH, Nikiforov A, Nikiforov I (2017) Combining iris and periocular biometric for matching visible spectrum eye images. Pattern Recogn Lett 91:11–16. Mobile Iris CHallenge Evaluation (MICHE-II)
11. Ahuja K, Islam R, Barbhuiya FA, Dey K (2016) A preliminary study of CNNs for iris and periocular verification in the visible spectrum. In: 2016 23rd International conference on pattern recognition (ICPR), pp 181–186
12. Ahuja K, Islam R, Barbhuiya FA, Dey K (2017) Convolutional neural networks for ocular smartphone-based biometrics. Pattern Recogn Lett 91:17–26. Mobile Iris CHallenge Evaluation (MICHE-II)
13. Alkassar S, Woo W-L, Dlay S, Chambers J (2016) Sclera recognition: on the quality measure and segmentation of degraded images captured under relaxed imaging conditions. IET Biometrics 6(4):266–275
14. Amjed N, Khalid F, Rahmat RWOK, Madzin HB (2018) Noncircular iris segmentation based on weighted adaptive hough transform using smartphone database. J Comput Theor Nanosci 15(3):739–743
15. Arandjelovic R, Zisserman A (2012) Three things everyone should know to improve object retrieval. In: 2012 IEEE conference on computer vision and pattern recognition. IEEE, pp 2911–2918
16. Arsalan M, Hong HG, Naqvi RA, Lee MB, Kim MC, Kim DS, Kim CS, Park KR (2017) Deep learning-based iris segmentation for iris recognition in visible light environment. Symmetry 9(11):263
17. Arsalan M, Naqvi RA, Kim DS, Nguyen PH, Owais M, Park KR (2018) Irisdensenet: robust iris segmentation using densely connected fully convolutional networks in the images by visible light and near-infrared light camera sensors. Sensors 18(5):1501

18. Barra S, Casanova A, Narducci F, Ricciardi S (2015) Ubiquitous iris recognition by means of mobile devices. Pattern Recogn Lett 57:66–73
19. Bowyer KW, Burge MJ (2016) Handbook of iris recognition. Springer, London
20. Clarke R (1994) Human identification in information systems: management challenges and public policy issues. Inf Technol People 7(4):6–37
21. Daugman J (2009) How iris recognition works. In: The essential guide to image processing. Elsevier, pp 715–739
22. Daugman JG (1993) High confidence visual recognition of persons by a test of statistical independence. IEEE Trans Pattern Anal Mach Intell 15(11):1148–1161
23. De Marsico M, Nappi M, Daniel R (2010) Is_is: Iris segmentation for identification systems. In: 2010 20th International conference on pattern recognition (ICPR). IEEE, pp 2857–2860
24. De Marsico M, Nappi M, Narducci F, Proença H (2018) Insights into the results of miche i-mobile iris challenge evaluation. Pattern Recogn 74:286–304
25. De Marsico M, Nappi M, Proença H (2017) Results from miche ii-mobile iris challenge evaluation ii. Pattern Recogn Lett 91:3–10
26. De Marsico M, Nappi M, Riccio D, Wechsler H (2015) Mobile iris challenge evaluation (miche)-i, biometric iris dataset and protocols. Pattern Recogn Lett 57:17–23
27. Dellana R, Roy K (2016) Data augmentation in CNN-based periocular authentication. Institute of Electrical and Electronics Engineers Inc., pp 141–145
28. Freire-Obregon D, Narducci F, Barra S, Castrill-Santana M (2018) Deep learning for source camera identification on mobile devices. Pattern Recogn Lett
29. Galdi C, Dugelay JL (2016) Fusing iris colour and texture information for fast iris recognition on mobile devices. In: 2016 23rd International conference on pattern recognition (ICPR). IEEE, pp 160–164
30. Galdi C, Dugelay J-L (2017) Fire: fast iris recognition on mobile phones by combining colour and texture features. Pattern Recogn Lett 91:44–51. Mobile Iris CHallenge Evaluation (MICHE-II)
31. Galdi C, Nappi M, Dugelay J-L (2015) Combining hardwaremetry and biometry for human authentication via smartphones. Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics), vol 9280, pp 406–416
32. Galdi C, Nappi M, Dugelay J-L (2016) Multimodal authentication on smartphones: combining iris and sensor recognition for a double check of user identity. Pattern Recogn Lett 82:144–153
33. Haindl M, Krupička M (2015) Unsupervised detection of non-iris occlusions. Pattern Recogn Lett 57:60–65
34. Hu Y, Sirlantzis K, Howells G (2015) Improving colour iris segmentation using a model selection technique. Pattern Recogn Lett 57:24–32
35. Huang B, Chen R, Zhou Q, Yu X (2018) Eye landmarks detection via two-level cascaded cnns with multi-task learning. Signal Proces: Image Commun 63:63–71
36. Jain AK, Dass SC, Nandakumar K (2004) Soft biometric traits for personal recognition systems. In: Biometric authentication. Springer, pp 731–738
37. Jain AK, Hong L, Pankanti S, Bolle R (1997) An identity-authentication system using fingerprints. Proc IEEE 85(9):1365–1388
38. Kauba C, Debiasi L, Uhl A (2018) Identifying the origin of iris images based on fusion of local image descriptors and PRNU based techniques, vol 2018-January. Institute of Electrical and Electronics Engineers Inc., pp 294–301
39. Lee MB, Hong HG, Park KR (2017) Noisy ocular recognition based on three convolutional neural networks. Sensors 17(12):2933
40. Li Y-H, Huang P-J (2017) An accurate and efficient user authentication mechanism on smart glasses based on iris recognition. Mobile Inf Syst
41. Liu N, Zhang M, Li H, Sun Z, Tan T (2016) Deepiris: learning pairwise filter bank for heterogeneous iris verification. Pattern Recogn Lett 82:154–161
42. Lukas J, Fridrich J, Goljan M (2006) Digital camera identification from sensor pattern noise. IEEE Trans Inf Forensics Secur 1(2):205–214

43. Ma L, Tan T, Wang Y, Zhang D (2003) Personal identification based on iris texture analysis. IEEE Trans Pattern Anal Mach Intell 12:1519–1533
44. Martin D, Fowlkes C, Tal D, Malik J (2001) A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings of eighth IEEE international conference on computer vision, 2001. ICCV 2001, vol 2. IEEE, pp 416–423
45. Ngiam J, Chen Z, Bhaskar SA, Koh PW, Ng AY (2011) Sparse filtering. In: Advances in neural information processing systems, pp 1125–1133
46. Nielsen J (2000) Security and human factors. Alertbox (November 2000). http://www.useit.com/alertbox/20001126.html
47. Oliva A, Torralba A (2001) Modeling the shape of the scene: a holistic representation of the spatial envelope. Int J Comput Vis 42(3):145–175
48. Patrick AS (2004) Usability and acceptability of biometric security systems. In: Financial cryptography, p 105
49. Pratt WK (2007) Digital image processing, 4th edn. Wiley, Hoboken, NJ
50. Proença H, Alexandre LA (2007) The nice. i: noisy iris challenge evaluation-part i. In: First IEEE international conference on biometrics: theory, applications, and systems, 2007. BTAS 2007. IEEE, pp 1–4
51. Proença H, Alexandre LA (2012) Introduction to the special issue on the recognition of visible wavelength iris images captured at-a-distance and on-the-move. Pattern Recogn Lett 33(8):963–964
52. Proenca H, Alexandre LA (2012) Toward covert iris biometric recognition: experimental results from the NICE contests. IEEE Trans Inf Forensics Secur 7(2):798–808
53. Proenca H, Filipe S, Santos R, Oliveira J, Alexandre LA (2010). The UBIRIS.v2: a database of visible wavelength iris images captured on-the-move and at-a-distance. IEEE Trans Pattern Anal Mach Intell 32(8):1529–1535
54. Radman A, Zainal N, Suandi S (2017) Automated segmentation of iris images acquired in an unconstrained environment using HOG-SVM and growcut. Digital Signal Process: Rev J 64:60–70
55. Raja KB, Raghavendra R, Vemuri VK, Busch C (2015) Smartphone based visible iris recognition using deep sparse filtering. Pattern Recogn Lett 57:33–42
56. Raja KB, Raghavendra R, Venkatesh S, Busch C (2017) Multi-patch deep sparse histograms for iris recognition in visible spectrum using collaborative subspace for robust verification. Pattern Recogn Lett 91:27–36. Mobile Iris CHallenge Evaluation (MICHE-II)
57. Rattani A, Derakhshani R (2017) Ocular biometrics in the visible spectrum: a survey. Image Vis Comput 59:1–16
58. Rattani A, Reddy N, Derakhshani R (2017) Gender prediction from mobile ocular images: a feasibility study. Institute of Electrical and Electronics Engineers Inc
59. Reddy N, Rattani A, Derakhshani R (2018) Ocularnet: deep patch-based ocular biometric recognition. In: 2018 IEEE International Symposium on Technologies for Homeland Security (HST). IEEE, pp 1–6
60. Roerdink JB, Meijster A (2000) The watershed transform: definitions, algorithms and parallelization strategies. Fundamenta informaticae 41(1, 2):187–228
61. Santos G, Grancho E, Bernardo MV, Fiadeiro PT (2015) Fusing iris and periocular information for cross-sensor recognition. Pattern Recogn Lett 57:52–59
62. Sasse MA (2007) Red-eye blink, bendy shuffle, and the yuck factor: a user experience of biometric airport systems. IEEE Secur Priv 5(3):78–81
63. Sasse MA, Brostoff S, Weirich D (2001) Transforming the ‘weakest link’—a human/computer interaction approach to usable and effective security. BT Technol J 19(3):122–131
64. Sun Z, Wang L, Tan T (2014) Ordinal feature selection for iris and palmprint recognition. IEEE Trans Image Process 23(9):3922–3934
65. Tan T, Zhang X, Sun Z, Zhang H (2012) Noisy iris image matching by using multiple cues. Pattern Recogn Lett 33(8):970–977

66. Wildes RP (1997) Iris recognition: an emerging biometric technology. Proc IEEE 85(9):1348–1363
67. Zhang H, Tian X, Deng X, Cao Y (2018) Batch process fault detection and identification based on discriminant global preserving kernel slow feature analysis. ISA Trans 79:108–126
68. Phillips PJ, Bowyer KW, Flynn PJ, Liu X, Scruggs WT (2008, September) The iris challenge evaluation 2005. In: 2008 IEEE second international conference on biometrics: theory, applications and systems. IEEE, pp 1–8
69. Phillips PJ, Scruggs WT, O'Toole AJ, Flynn PJ, Bowyer KW, Schott CL, Sharpe M (2009) FRVT 2006 and ICE 2006 large-scale experimental results. IEEE Trans Pattern Anal Mach Intell 32(5):831–846

Chapter 5
Super-resolution for Selfie Biometrics: Introduction and Application to Face and Iris

Fernando Alonso-Fernandez, Reuben A. Farrugia,
Julian Fierrez and Josef Bigun

Abstract Biometrics research is heading towards enabling more relaxed acquisition conditions. This has effects on the quality and resolution of acquired images, severely affecting the accuracy of recognition systems if not tackled appropriately. In this chapter, we give an overview of recent research in super-resolution reconstruction applied to biometrics, with a focus on face and iris images in the visible spectrum, two prevalent modalities in selfie biometrics. After an introduction to the generic topic of super-resolution, we investigate methods adapted to cater for the particularities of these two modalities. By experiments, we show the benefits of incorporating super-resolution to improve the quality of biometric images prior to recognition.

The lack of resolution has a negative impact on the performance of image-based biometrics. Many applications which are becoming ubiquitous in mobile devices do not operate in a controlled environment, and their performance significantly drops due to the lack of pixel resolution, since it decreases the amount of information available for recognition [41].

While many generic super-resolution techniques have been studied to restore low-resolution images for biometrics [54, 73], the results obtained are not always as desired. Those generic super-resolution methods are usually aimed to enhance the visual appearance of the scene. However, producing an overall visual enhancement

F. Alonso-Fernandez (✉) · J. Bigun
School of Information Technology (ITE), Halmstad University,
Box 823, 30118 Halmstad, Sweden
e-mail: feralo@hh.se

J. Bigun
e-mail: josef.bigun@hh.se

R. A. Farrugia
Department of Communications and Computer Engineering (CCE),
University of Malta, Msida, Malta
e-mail: reuben.farrugia@um.edu.mt

J. Fierrez
Escuela Politecnica Superior, Universidad Autonoma de Madrid, 28049 Madrid, Spain
e-mail: julian.fierrez@uam.es

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision
and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_5

of biometric images does not necessarily correlate with a better recognition performance [6, 29]. Such techniques are designed to restore generic images and therefore do not exploit the specific structure found in biometric images (e.g. iris or faces), which causes the solution to be sub-optimal [22]. For this reason, super-resolution techniques have to be adapted to cater for the particularities of images from a specific biometric modality [8].

In recent years, there has been an increased interest in the application of super-resolution to different biometric modalities, such as face iris, gait or fingerprint [60]. This chapter presents an overview of recent advances in super-resolution reconstruction of face and iris images, which are the two prevalent modalities in selfie biometrics. We also provide experimental results using several state-of-the-art reconstruction algorithms, demonstrating the benefits of using super-resolution to improve the quality of face and iris images prior to classification. In the reported experiments, we study the application of super-resolution to face and iris images captured in the visible range, using experimental set-ups that represent well the selfie biometrics scenario. The chapter begins with a general introduction to image resolution, including the usual mathematical formulation, a brief taxonomy of super-resolution methods, and performance metrics. We then focus on face biometrics, describing recent super-resolution methods adapted for this biometrics including experimental results. Another section on iris super-resolution follows with a parallel structure. The chapter ends with a summary and an outlook of future trends.

5.1 Image Super-Resolution

The performance of biometric recognition systems and the quality perceived by the human visual system (HVS) is significantly affected by the resolution of the image. These images can be up-scaled using classical interpolation schemes used in several commercial software such as bilinear and bicubic interpolation [31]. These methods use kernels that assume that the image data is either spatially smooth or band-limited and usually reconstruct blurred images [74]. More sophisticated interpolation methods were proposed in [11, 71] that manage to restore sharper images at the expense of generating visual artefacts in texture-less regions of the image. While more advanced interpolation schemes manage to restore sharper images, they fail to reliably restore texture detail which is important for biometric recognition systems.

Several researchers have proposed more advanced techniques that recover the lost high-frequency information. These methods usually formulate the problem using the following acquisition model

$$\mathbf{X} = \mathbf{D}\mathbf{B}\mathbf{Y} + \boldsymbol{\eta} \tag{5.1}$$

where $\mathbf{X}$ is the observed low-resolution image, $\mathbf{B}$ is the blurring kernel, $\mathbf{D}$ is the down-sampling matrix, $\boldsymbol{\eta}$ represents additive noise and $\mathbf{Y}$ is the unknown high-resolution image to be estimated.

Existing super-resolution methods can be categorized into two groups: (i) *Reconstruction-based* super-resolution techniques, which exploit the redundancies present in images and videos to estimate and restore an image, and (ii) *Learning-based* super-resolution methods, which treat the problem as an inverse problem and learn a mapping relation between the low- and high-resolution images. More detail about each category is provided in the following subsections, while a comprehensive survey can be found in [54].

5.1.1 Reconstruction-Based Methods

Reconstruction-based algorithms try to address the aliasing artifacts that are present in the observed low-resolution images due to the under-sampling process. Iterative back projection (IBP) methods [38, 39] use the acquisition model defined in Eq. (5.1). These methods first register a sequence of low-resolution images over the high-resolution grid which are then averaged to estimate the high-resolution image. IBP is then used to refine that initial solution. To facilitate convergence and increase robustness against outliers, the authors in [23, 82] regularize the objective function using either smoothness or sparse constraints.

These methods were later on extended by considering different fusion kernels and including a de-blurring filter as a post-process, e.g. using the Wiener Filter as suggested in [31]. The authors in [24, 25] have shown that the median fusion of the registered low-resolution images is equivalent to the maximum-likelihood estimation and results in a robust super-resolution algorithm if the motion between the low-resolution frames is translational. Different data fusion techniques [26] based on adaptive averaging [65], Adaboost classification [70] and SVD-based filters [53] were also considered. Probabilistic-based super-resolution techniques based on the maximum-likelihood (ML) [14, 68] and maximum a-posteriori (MAP) [32] were proposed to estimate the high-resolution frame.

More recently, a framework that extends reconstruction-based super-resolution methods for the single image super-resolution problem was proposed in [30]. This method is based on the observation that patches in a natural image tend to reoccur many times inside an image, both within the same scale as well as across different scales.

On the other hand, Lin et. al. [49] have derived the theoretical limits of reconstruction-based super-resolution methods and proved that they can only achieve low magnification factors (≤ 2). Moreover, these methods (except for the work in [30]) are only applicable for video sequences, i.e. they require several low-resolution images as input, and in general, they fail to restore dynamic non-rigid objects such as faces.

Due to the limitations of reconstruction-based super-resolution, the research community is increasingly paying more attention to learning-based super-resolution methods, which can recover more texture detail and achieve higher magnification factors. They also have the advantage of only needing one image as input.

5.1.2 Learning-Based Methods

The seminal work of Freeman [28] presented the first example-based (*a.k.a.* learning-based) super-resolution algorithm. This class of methods employs a couple dictionary of low- and corresponding high-resolution patches which are constructed by collecting collocated patches from a set of low- and high-resolution training images. Figure 5.1 illustrates the principle of how the low-resolution **L** and high-resolution **H** dictionaries are constructed. The authors in [28] then proposed to subdivide the input image into low-resolution patches that are traversed in raster-scan order. At each step, a low-frequency patch is selected by a nearest neighbour search from the low-resolution dictionary **L**. The high-resolution patch is then estimated using the collocated patch in the high-resolution dictionary **H**. Markov Random Fields are then used to enforce smoothness across neighbouring patches. The reconstructed high-resolution patches are then stitched together to form the high-resolution image.

The authors in [13] observed that small patches from low- and high-resolution images form manifolds with similar local geometry in two distinct spaces. They then use local linear embedding (LLE) to find the k-closest neighbours from **L** to the ith low-resolution patch $\mathbf{x}_i$ to form the sub-dictionary $\mathbf{L}_k$. The reconstruction weights **w** are then computed using the following optimization problem

$$\mathbf{w} = \arg\min_{\mathbf{w}} ||\mathbf{x}_i - \mathbf{L}_k\mathbf{w}||_2^2 \quad \text{subject to} \quad \sum_j w_j = 1 \tag{5.2}$$

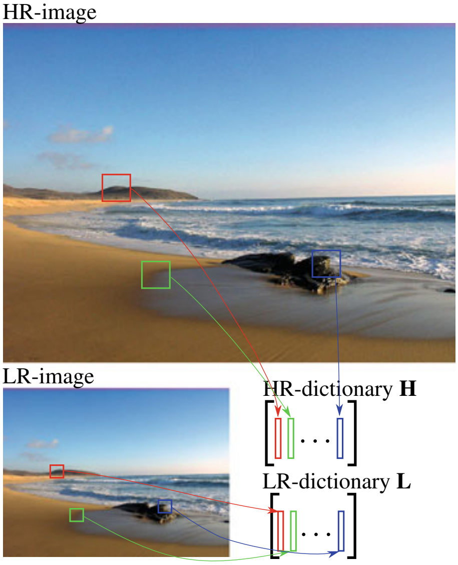


Fig. 5.1 Dictionary construction for a learning-based super-resolution algorithm

which has a closed form solution. The high-resolution patch $\tilde{\mathbf{y}}_i$ is then reconstructed using

$$\tilde{\mathbf{y}}_i = \mathbf{H}_k \mathbf{w} \tag{5.3}$$

where $\mathbf{H}_k$ correspond to the k column vectors from $\mathbf{H}$ that correspond to the k-closest neighbours on the low-resolution manifold. Several researchers have proposed different ways of estimating the combination weights $\mathbf{w}$, the most notorious one is to pose a sparsity constraint on the weights as done in [79]

$$\mathbf{w} = \arg\min_{\mathbf{w}} ||\mathbf{x}_i - \mathbf{L}\mathbf{w}||_2^2 \quad \text{subject to} \quad ||\mathbf{w}||_1 \tag{5.4}$$

that can be computed in polynomial time using sparse coding solvers and is capable to outperform the neighbour-embedding scheme [13]. Later on, the same group has shown in [80] that performance can be further improved using dictionary learning techniques that jointly train the low- and high-resolution dictionaries to generate a more compact representation of the patch pairs which simply sample a large amount of image patch pairs as shown in Fig. 5.1.

The authors in [19] have shown that sparse representations are affected by the distortions present in the low-resolution image and are therefore not accurate enough to faithfully reconstruct the original image. They then reformulate the sparse coding problem in (5.4) as

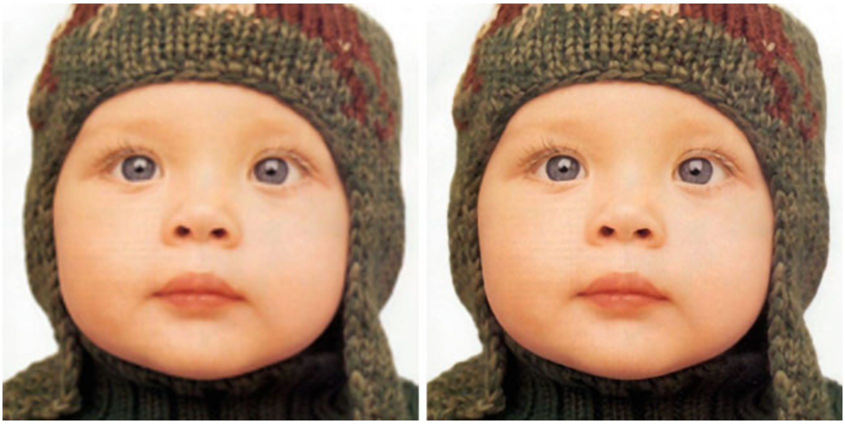
$$\mathbf{w} = \arg\min_{\mathbf{w}} ||\mathbf{x}_i - \mathbf{L}\mathbf{w}||_2^2 \quad \text{subject to} \quad ||\mathbf{w}||_1 \quad \text{and} \quad \sum_j (w_j - \beta_j)^2 \leq \varepsilon \tag{5.5}$$

where β is estimated from the sparse coding coefficients of neighbouring patches.

Deep convolutional neural networks (DCNN) were investigated recently for the generic super-resolution task. In [18], the authors present a shallow network consisting of just three convolutional layers, providing substantial improvement over sparse coding-based super-resolution methods. This model poses the super-resolution problem as a regression problem and uses a DCNN to model a function $f(\mathbf{X} : \theta)$ that minimizes the following loss function

$$L(\theta) = \sum_j f((\mathbf{X}_j : \theta) - \mathbf{Y}_j)^2 \tag{5.6}$$

where $\mathbf{X}_j$ and $\mathbf{Y}_j$ represent a set of low- and corresponding high-resolution training images, j is an index and θ are the hyperparameters of the network. More recently, very deep architectures were proposed in [45, 48] which employ deeper architectures and residual learning and are reported to provide state-of-the-art performance. The results in Fig. 5.2 show the performance of VDSR [45] against bicubic interpolation where it can be clearly seen that VDSR is able to recover sharper images.



Bicubic VRSD

Fig. 5.2 Comparing the performance of a very deep CNN (VDSR) against bicubic interpolation

5.1.3 Performance Metrics

To evaluate the performance of super-resolution methods, the peak signal-to-noise ratio (PSNR) and the structural similarity (SSIM) index between the enhanced and the corresponding high-resolution reference images are usually employed [73].

The PSNR is a measure of the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. The signal in this case is the reference high-resolution image $\mathbf{Y}$, and the noise is the error introduced in its estimation $\tilde{\mathbf{Y}}$ by the reconstruction algorithm. Considering greyscale images of size $N \times M$ and grey values in the [0, 255] range, it is defined (in dBs) as:

$$\text{PSNR}(\mathbf{Y}, \tilde{\mathbf{Y}}) = 10 \log_{10} \left[\frac{255^2}{\text{MSE}(\mathbf{Y}, \tilde{\mathbf{Y}})} \right] \tag{5.7}$$

where $\text{MSE}(\mathbf{Y}, \tilde{\mathbf{Y}})$ is the mean squared error given by

$$\text{MSE}(\mathbf{Y}, \tilde{\mathbf{Y}}) = \left[\frac{1}{NM} \sum_{i=1}^{M} \sum_{j=1}^{N} \left| \mathbf{Y}(i, j) - \tilde{\mathbf{Y}}(i, j) \right|^2 \right] \tag{5.8}$$

A higher PSNR generally indicates that the reconstruction is of higher quality. If the two images are identical, $\text{MSE}(\mathbf{Y}, \tilde{\mathbf{Y}}) = 0$, in whose case $\text{PSNR}(\mathbf{Y}, \tilde{\mathbf{Y}}) = \infty$. The PSNR is an estimation of the absolute error between two images. The SSIM index, on the other hand, is a perception-based model that considers image degradation as a perceived change in structural information. This is achieved by using first- and second-order statistics of grey values on local image windows. It is computed on various windows of an image. Given two collocated image windows $\mathbf{y}$ and $\tilde{\mathbf{y}}$, the SSIM index is computed as

$$\mathrm{SSIM}\left(\mathbf{y}, \tilde{\mathbf{y}}\right) = \frac{\left(2\mu_{\mathbf{y}}\mu_{\tilde{\mathbf{y}}} + c_1\right)\left(2\sigma_{\mathbf{y}\tilde{\mathbf{y}}} + c_2\right)}{\left(\mu_{\mathbf{y}}^2 + \mu_{\tilde{\mathbf{y}}}^2 + c_1\right)\left(\sigma_{\mathbf{y}}^2 + \sigma_{\tilde{\mathbf{y}}}^2 + c_2\right)} \tag{5.9}$$

where the parameter $\mu_{\mathbf{y}}$ ($\mu_{\tilde{\mathbf{y}}}$) is the average grey value of $\mathbf{y}$ ($\tilde{\mathbf{y}}$), the parameter $\sigma_{\mathbf{y}}$ ($\sigma_{\tilde{\mathbf{y}}}$) is the variance of the grey values of $\mathbf{y}$ ($\tilde{\mathbf{y}}$), and $\sigma_{\mathbf{y}\tilde{\mathbf{y}}}$ is the covariance of $\mathbf{y}$ and $\tilde{\mathbf{y}}$. By default, $c_1 = (0.01 * 255)^2$ and $c_2 = (0.03 * 255)^2$ [78]. Also, the window size is of 11×11, which is weighted with a circular Gaussian filter of standard deviation 1.5 before calculating local statistics. The SSIM index is computed for all pixels of the image, which are then averaged to obtain the SSIM index of the overall image. The SSIM index is a decimal value between -1 and 1, and value 1 is only reachable in the case of two identical images.

The use of super-resolution techniques in general applications is aimed at improving the overall visual perception and appearance. In biometrics, however, the aim is to improve the recognition performance [60]. While PSNR or SSIM are the standard metrics widely used in the super-resolution literature, they are not necessarily good predictors of the recognition accuracy. Human and machine evaluations of image quality may differ, and human judgement may not be relevant to biometric algorithms [6]. For this reason, reporting recognition performance using enhanced images is necessary to evaluate the goodness of the reconstruction algorithm in a biometric context.

5.2 Face Super-Resolution

In their seminal work in 2000, Baker and Kanade [7] exploited the fact that human face images are a relatively small subset of natural scenes and introduced the concept of class-based super-resolution, i.e. only facial images are used to learn the coupled dictionaries $\mathbf{L}$ and $\mathbf{H}$. This method employs a pyramid-based algorithm to learn a prior on the derivative of the high-resolution facial images as a function of the spatial location in the image and the information from higher levels of the pyramid and the solution is derived using the MAP algorithm. The authors in [35] observed that similar faces share similar local structure and synthesize missing pixels using a linear combination of spatially neighbouring pixels. This method was extended in [47] where they exploit the sparse nature of the pixel structure. However, the performance of these reconstruction-based methods significantly degrades when considering large magnification factors where the local pixel structure is degraded.

5.2.1 Face Eigentransformation

Face representation models were used in [12, 63, 77] to derive a set of low- and high-resolution prototypes. The low-resolution face image is reconstructed using a weighted combination of low-resolution prototypes, and the learned weights are used to combine the high-resolution prototypes to synthesize the high-resolution face image. To explain this principle, we take the classical Eigentransformation method [77] which was used as a baseline in several studies. The low-resolution and high-resolution mean faces ($\mathbf{m}_L$ and $\mathbf{m}_H$, respectively) are computed as

$$\hat{\mathbf{m}}_L = \frac{1}{M}\sum_{i=1}^{M}\mathbf{L}_i \quad \text{and} \quad \hat{\mathbf{m}}_H = \frac{1}{M}\sum_{i=1}^{M}\mathbf{H}_i \tag{5.10}$$

where M is the number of training faces and the notation $\mathbf{K}_i$ denotes the ith column vector of matrix $\mathbf{K}$. The coupled dictionaries are then centred using

$$\bar{\mathbf{L}} = \mathbf{L} - \mathbf{m}_L \quad \text{and} \quad \bar{\mathbf{H}} = \mathbf{H} - \mathbf{m}_H \tag{5.11}$$

Given an input low-resolution image $\mathbf{X}$, it can be approximated using a weighted combination of centred faces using

$$\tilde{\mathbf{X}} = \bar{\mathbf{L}}\mathbf{w} + \mathbf{m}_L \tag{5.12}$$

where

$$\mathbf{w} = \mathbf{V}_L \Lambda_L^{-\frac{1}{2}} \mathbf{E}^T (\mathbf{X} - \mathbf{m}_L) \tag{5.13}$$

where $\mathbf{V}_L$ is the eigenvector matrix, Λ_L is the eigenvalue matrix and $\mathbf{E}$ is the eigenface matrix which are derived by computing PCA on the covariance matrix $\mathbf{C}_L = \bar{\mathbf{L}}^T\bar{\mathbf{L}}$. The reconstruction of the high-resolution face image is done by simply replacing the low-resolution dictionary $\bar{\mathbf{L}}$ with the high-resolution dictionary $\bar{\mathbf{H}}$ and the low-resolution mean face $\mathbf{m}_L$ with the high-resolution mean face $\mathbf{m}_H$ in (5.12), which is therefore computed using

$$\tilde{\mathbf{Y}} = \bar{\mathbf{H}}\mathbf{w} + \mathbf{m}_H \tag{5.14}$$

The above methods are able to hallucinate missing information by exploiting the global facial structure. Nevertheless, the faces restored using these methods are generally noisy and their quality is usually inferior to bicubic interpolation. This is mainly attributed to the fact that the dimension of the face image is much larger than the number of training examples which makes the dictionaries undercomplete.

More recently, the authors in [51] exploited the structure of the face and constructed position-dependent dictionaries, as shown in Fig. 5.3. Face images are first aligned using affine transformation such that the eyes and mouth centres are aligned and then they are divided into overlapping patches. Then they construct a coupled dictionary for each patch-position. In the example in Fig. 5.3, the high-resolution

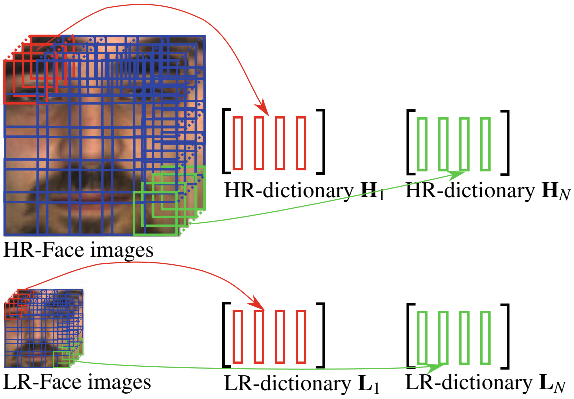

HR-Face images
HR-dictionary $\mathbf{H}_1$
HR-dictionary $\mathbf{H}_N$
LR-Face images
LR-dictionary $\mathbf{L}_1$
LR-dictionary $\mathbf{L}_N$

Fig. 5.3 Dictionary construction using the position-patch principle

dictionary $\mathbf{H}_1$ (marked in red) consists of a vectorized representation of top-left position of all the N high-resolution images while the corresponding low-resolution dictionary $\mathbf{L}_1$ is composed of the vectorized representations of the N low-resolution images. This simple extension reduces the dimensionality of the problem and reduces the possibility of over fitting. During testing, the low-resolution input image $\mathbf{X}$ is dissected into a set of overlapping patches that we shall denote as $\mathbf{x}_j$ where $j \in [1, N]$ represents the patch-position. For each position-patch j, the authors use the coupled low- and high-resolution dictionaries $\mathbf{L}_j$ and $\mathbf{H}_j$, respectively. Each patch is restored independently and then stitched together by averaging overlapping pixels.

The authors in [51] proposed to formulate the restoration of a patch as a least squares problem

$$\mathbf{w}_j = \arg\min_{\mathbf{w}_j} ||\mathbf{H}_j - \mathbf{L}_j\mathbf{w}_j||_2^2 \tag{5.15}$$

which has the following closed form solution

$$\mathbf{w}_j = \left(\mathbf{L}_j^T\mathbf{L}_j\right)^{\dagger} \mathbf{L}_j^T\mathbf{H}_j \tag{5.16}$$

where † stands for the pseudo-inverse operator. Several researchers have extended the position-patch method using different objective functions. The authors in [44] have formulated the weight estimation problem using sparse coding

$$\mathbf{w}_j = \arg\min_{\mathbf{w}_j} ||\mathbf{H}_j - \mathbf{L}_j\mathbf{w}_j||_2^2 \quad \text{subject to} \quad ||\mathbf{w}_j||_1 \tag{5.17}$$

which enforces the combination weights $\mathbf{w}_j$ to be sparse. This regularization allows deriving sharper facial images and is more robust to outliers.

5.2.2 Local Iterative Neighbour Embedding

One drawback of the methods of Sect. 5.2.1 is that they assume that low- and high-resolution manifolds have similar local geometrical structure. Reconstruction weights are estimated on the low-resolution manifold, and they are simply transferred to the high-resolution manifold. However, the low-resolution manifold is distorted by the one-to-many relationship between low- and high-resolution patches [46, 72]. Therefore, the reconstruction weights estimated on the low-resolution manifold do not necessarily correlate with the actual weights needed to reconstruct the unknown high-resolution patch on the high-resolution manifold.

Motivated by this observation, the authors in [36, 46] derive a pair of projection matrices that can be used to project both low- and high-resolution patches on a common coherent sub-space. However, the dimension of the coherent sub-space is equal to the lowest rank of the low- and high-resolution dictionary matrices. Therefore, the projection from the coherent sub-space to the high-resolution manifold is ill-conditioned.

This ill-conditioning is overcome in the locality-constrained iterative neighbour embedding (LINE) method presented in [42] as follows. They first estimate the high-resolution patch $\mathbf{v}_{0,0}$ by up-scaling the low-quality patch $\mathbf{x}_j$ using bicubic interpolation and initialize the intermediate dictionary as $\mathbf{L}_j^{\{0\}} = \mathbf{L}_j$. This iterative method has an outer loop indexed by $b \in [0, B-1]$ and an inner loop indexed by $c \in [0, C-1]$. For every iteration of the inner loop, the supports $\mathbf{s}$ (i.e. the column vectors) of $\mathbf{H}_j$ are derived as the k-nearest neighbours of $\mathbf{v}_{b,c}$. The combination weights are then computed using

$$\mathbf{w} = \arg\min_{\mathbf{w}} ||\mathbf{x}_j - \mathbf{L}_j^{\{b\}}(\mathbf{s})\mathbf{w}||_2^2 + \tau||\mathbf{d}(\mathbf{s}) \odot \mathbf{w}||_2^2 \tag{5.18}$$

where τ is a regularization parameter, $\odot$ is the element-wise multiplication operator and $\mathbf{d}(\mathbf{s})$ measures the Euclidean distance between the restored patch $\mathbf{v}_{b,c}$ and the k-nearest neighbours column vectors from the high-resolution dictionary $\mathbf{H}_j$. This optimization problem has an analytical solution and the high-resolution patch is updated using

$$\mathbf{v}_{b,c+1} = \mathbf{H}_j(\mathbf{s})\mathbf{w} \tag{5.19}$$

Once all iterations of the inner loop are completed, the intermediate dictionary $\mathbf{L}_j^{\{b+1\}}$ is updated using a leave-one-out methodology as described in [42] and the inner loop is repeated. The final estimate of the high-resolution patch is then simply $\mathbf{v}_{B,C-1}$.

5.2.3 Linear Model of Coupled Sparse Support

While the method of Sect. 5.2.2 iteratively updates the low-resolution dictionary to restore the geometrical structure in the low-resolution manifold, it cannot guarantee to converge to an optimal solution. Farrugia et al. [22] later presented the linear model of coupled sparse support (LM-CSS) which learns linear models based on the local geometrical structure on the high-resolution manifold rather than the low-resolution manifold. For this, in a first step, the low-resolution patch is used to derive a globally optimal estimate of the high-resolution patch. This is equivalent to solving the following problem

$$\boldsymbol{\Phi} = \arg\min_{\boldsymbol{\Phi}} ||\mathbf{H}_j - \boldsymbol{\Phi}\mathbf{L}_j||_2^2 \quad \text{subject to} \quad ||\boldsymbol{\Phi}||_2^2 \leq \varepsilon \tag{5.20}$$

which can be solved using Ridge regression. This approximated solution is close in Euclidean space to the ground truth but is generally smooth and lacks the texture details needed by state-of-the-art face recognizers. The authors then search for the sparse support that best estimates the first approximated solution on the high-resolution manifold where the geometric structure of manifold is intact. The derived support is then used to extract the atoms from the coupled low- and high-resolution dictionaries $\mathbf{L}_j$ and $\mathbf{H}_j$ that are most suitable to learn an up-scaling function for every position-patch. The second step reformulates the problem as in Equation (5.20), where only a subset of the column vectors, defined by the support, are used to find the solution. This work also demonstrates that sparsity leads to sharper solutions and generally results in higher recognition accuracies. The same authors have also demonstrated that these super-resolution techniques can be applied to restore compressed low-resolution facial images [21].

5.2.4 Results

In the experiments reported here, we consider a composite dataset which includes images from both colour FERET and Multi-PIE datasets, where only frontal facial images were considered. One image per subject was randomly selected, resulting in a dictionary of 1203 facial images. The gallery for the evaluation included images from both FRGC-V2 (controlled environment) and MEDS datasets. One unique image per subject was randomly selected, providing a gallery of 889 facial images. The probe images were taken from the FRGC (uncontrolled environment), where two images per subject were included, resulting in 930 probe images. All the images were registered using affine transformation computed on landmark points of the eyes and mouth centres such that the distance between the eyes is set to 40 pixels. The probe and low-resolution dictionary images were down-sampled to the desired scale using MATLAB's *imresize* function.

Table 5.1 Summary of the quality analysis results using the PSNR and SSIM metrics on the FRGC dataset

SR Method	Inter eye distance							
	8		10		15		20	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bicubic	24.0292	0.6224	26.2024	0.7338	25.2804	0.7094	28.6663	0.8531
Eigentransformation [77]	24.3958	0.6496	26.8645	0.7504	24.9374	0.6724	27.7883	0.7892
Neighbour embedding [13]	26.9987	0.7533	27.9560	0.7973	29.9892	0.8714	31.6301	0.9122
Position-Patches [51]	27.3044	0.7731	28.2906	**0.8145**	30.1887	0.8785	31.7192	0.9143
Sparse Position-Patches [44]	27.2500	0.7666	28.2219	0.8100	30.1290	0.8767	31.7162	0.9146
LINE [42]	27.0927	0.7591	28.0253	0.8009	30.0471	0.8727	31.6970	0.9131
LM-CSS [22] ($k = 50$)	27.1307	0.7679	28.1078	0.8093	30.0240	0.8761	31.6875	0.9139
LM-CSS [22] ($k = 150$)	**27.4866**	**0.7802**	**28.4200**	0.8009	**30.3431**	**0.8845**	**31.9610**	**0.9209**

The best result of each column is marked in bold

The results in Tables 5.1 and 5.2 evaluate the performance of different face super-resolution techniques mentioned in this chapter in terms of both quality (PSNR and SSIM) and recognition performance (rank-1 and Area Under the Curve), respectively. It can be seen that the global Eigentransformation method [77] most of the time performs worse than bicubic. This can be explained by the fact that while it reconstructs a face that is visually more pleasing than the ones obtained using bicubic interpolation, it fails to reliably recover the local texture detail (see example images in Fig. 5.4). On the other hand, the remaining patch-based schemes outperform bicubic interpolation in terms of both quality and recognition performance. The VGG-Face CNN face recognition system (DeepFaces) is also found to be particularly fragile, performing considerably worse than LBP for any resolution. For example, with a magnification factor of just 2 (Inter Eye distance = 20), its rank-1 accuracy is equal or below 40% for any reconstruction technique. It can also be noticed that while position-patch [51] achieves higher PSNR and SSIM compared to sparse position-patch [44], LINE [42] and LM-CSS with $k = 50$ [22], it does not perform well in terms of face recognition performance. The authors in [22] show experimentally using different face recognizers that the PSNR and SSIM metrics do not correlate well with the recognition performance since they are biased to provide high scores to blurred images. It can be seen in Fig. 5.4 that the images restored via position-patches are more blurred, which harm the recognition performance. They also showed that sparse-based methods [22, 42, 44] are able to better preserve the texture detail and thus are able to achieve higher recognition performance. The results in Fig. 5.4 also show that the LINE method generally reconstructs sharp facial images, although they

Table 5.2 Summary of the Rank-1 recognition performance and Area Under the Curve (AUC) metric using the LBP [1] and DeepFaces [64] face recognition algorithm on the FRGC dataset

SR method	Comparator	Inter eye distance							
		8		10		15		20	
		Rank-1	AUC	Rank-1	AUC	Rank-1	AUC	rank-1	AUC
Bicubic	LBP	0.3065	0.9380	0.5032	0.9598	0.6065	0.9708	0.7054	0.9792
	DeepFaces	0.0000	0.5296	0.0000	0.5337	0.0258	0.6223	0.1903	0.7157
Eigentransformation [77]	LBP	0.2559	0.9390	0.4516	0.9554	0.5624	0.9633	0.6495	0.9688
	DeepFaces	0.0151	0.5794	0.0194	0.5954	0.0398	0.6187	0.1226	0.6612
Neighbour embedding [13]	LBP	0.5548	0.9635	0.6398	0.9712	0.7215	0.9795	0.7559	0.9830
	DeepFaces	0.0151	0.5940	0.0602	0.6290	0.2086	0.6970	0.4075	0.7562
Sparse Position Patches [44]	LBP	0.5677	0.9649	0.6441	0.9721	0.7247	0.9803	0.7570	0.9830
	DeepFaces	0.0161	0.5870	0.0419	0.6198	0.1624	0.6880	0.3796	0.7553
Position Patches [51]	LBP	0.4699	0.9588	0.5849	0.9675	0.6849	0.9782	0.7312	0.9812
	DeepFaces	0.0161	0.5914	0.0398	0.6201	0.1785	0.6878	0.3559	0.7408
LINE [42]	LBP	0.5925	0.9647	0.6559	0.9714	0.7323	**0.9804**	0.7677	**0.9833**
	DeepFaces	0.0312	0.6036	0.07 10	0.6385	0.2161	0.7050	0.4172	0.7630
LM-CSS [22] ($k = 50$)	LBP	**0.6032**	**0.9658**	**0.6581**	**0.9722**	**0.7398**	0.9798	**0.7742**	**0.9833**
	DeepFaces	0.0172	0.5874	0.0484	0.6293	0.1914	0.7015	0.4022	0.7609
LM-CSS [22] ($k = 150$)	LBP	0.5452	0.9644	0.6344	0.9710	**0.7398**	0.9801	0.7602	0.9831
	DeepFaces	0.0151	0.5890	0.0527	0.6291	0.2108	0.7011	0.3828	0.7578

The best result of each column is marked in bold

tend to be noisy. This noise does not seem to harm the recognition performance, but it may make it hard for a human to recognize a person from such noisy images.

One of the major problems in these methods is that they assume that the face images are aligned and cannot be applied directly to restore faces with random pose and orientation. The authors in [22] presented a simple method that registers the faces in the dictionary where a set of landmark points are used to register the dataset with the low-resolution image using piecewise affine transformation. Any face super-resolution method described in this chapter can then be used to restore faces with unconstrained poses. However, this approach is computationally intensive and it is difficult for a user to accurately mark the landmark points on very low-resolution images. Following the success of deep learning for generic super-resolution, the authors in [10] applied deep learning to directly restore facial images at arbitrary poses without the need for pre-registration. The main advantage of this method is that it is very fast to compute, it does not need human interaction and it is able to restore the whole head including the hair region. Nevertheless, while the results presented in the paper are promising, they were not assessed in terms of face recognition performance.

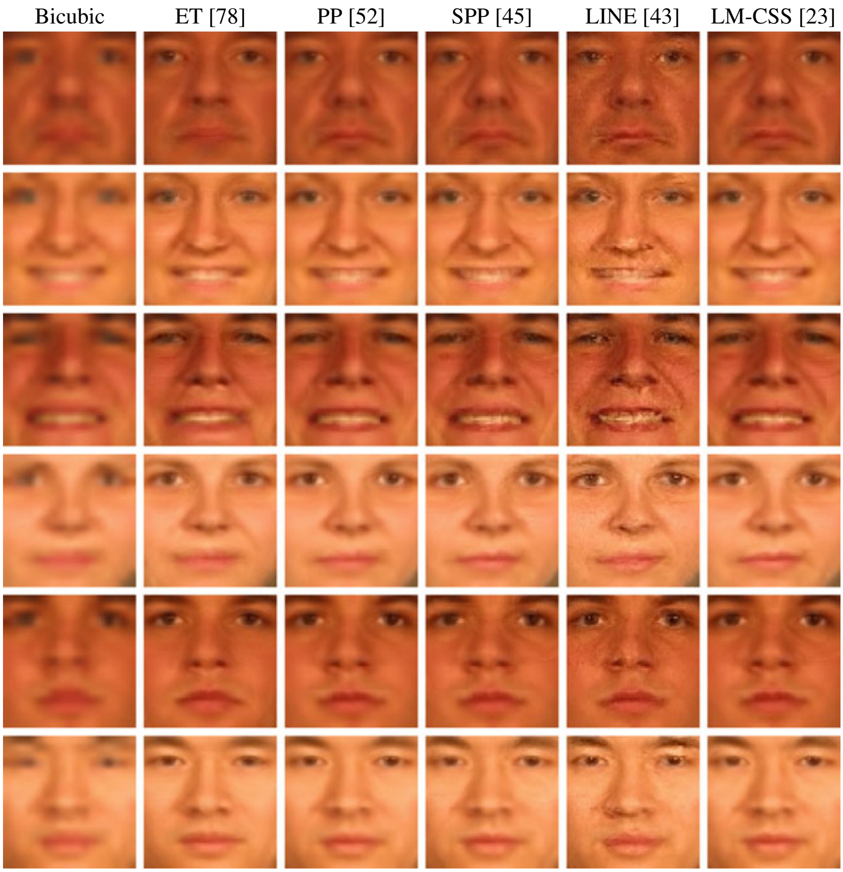


Fig. 5.4 Super-resolved face images using different face super-resolution techniques with a magnification factor of ×4 with an inter-eye distance of 10 pixels

5.3 Iris Super-Resolution

Super-resolution was introduced to the iris modality in 2003 by Huang et al. [37]. This method learns the probabilistic relation between different frequency bands, which is used to predict the missing high-frequency information of low-resolution images. Reconstruction-based methods for iris started in 2006 with the work of Barnard et al. [9], where they employed a multi-lens imaging hardware system to capture multiple iris images. Reconstruction was done by modelling the least square inverse problem associated with Eq. (5.1). Later, Fahmy [20] proposed to estimate high-resolution images using an auto-regressive model that fuses a sequence of low-resolution iris images. While these two works super-resolve the original iris image, most of the existing reconstruction-based methods employ the unwrapped polar image as input.

Several polar images are aligned and combined pixel-wise to obtain a reconstructed image. Given a set of N polar iris images $\mathbf{X}_i$, the super-resolved image $\tilde{\mathbf{Y}}$ is estimated as

$$\tilde{\mathbf{Y}}(x, y) = \frac{\sum_{i=1}^{N} w_i \mathbf{X}_i(x, y)}{\sum_{i=1}^{N} w_i} \tag{5.21}$$

where $\tilde{\mathbf{Y}}(x, y)$ is the intensity value of at pixel (x, y) of the super-resolved image, $\mathbf{X}_i(x, y)$ is the intensity value at the same location of the input image i, and w_i are the combination weights. The weights can be derived to simply compute the mean or median of the pixel values, as in [33, 43, 55]. Other studies have proposed to incorporate quality measures [34, 56, 58, 62], so more weight is given to frames with higher quality [27]. Recent reconstruction-based studies have proposed the use of Gaussian process regression (GPR), enhanced iterated back projection (EIBP) [17] and total variation regularization algorithms [16] to super-resolve polar frames.

Regarding learning-based methods, several algorithms have been proposed to learn the mapping between low- and high-resolution images, for example, multilayer perceptrons [69], Markov networks [50] or Bayesian modelling [2]. Some works have also proposed to super-resolve images in the feature space, instead of the pixel domain. This strategy has been followed with Eigeniris features [57] (similar to Eigenfaces proposed in [75]) and with the popular iris Gabor features [59, 61]. Recent studies also make use of convolutional neural networks, such as [67, 81].

Despite the now extensive literature on iris super-resolution, the majority of works have employed near-infrared data, which is the prevalent illumination in commercial systems. In the experiments reported in the present chapter, we study the application of super-resolution to iris images captured in the visible range using various smartphones, using an experimental setup that represents well the selfie biometrics scenario.

5.3.1 Iris Eigen-Patches

The work [4] proposed the use of principal component analysis (PCA) Eigen-transformation of local image patches to compute a reconstructed iris image. The technique is inspired by the system of [15] for face images. It employs the Eigen-transformation method defined by Eqs. (5.10)–(5.14) [77], but applied to overlapping patches, as shown in Fig. 5.5. The iris images are first resized via bicubic interpolation to have the same iris radius, and then aligned by extracting a square region around the pupil centre. Images are then divided into overlapping patches, and a coupled dictionary is constructed for each patch-position. Each patch is reconstructed separately, and a preliminary reconstructed image $\tilde{\mathbf{Y}}'$ is obtained by averaging the overlapping regions. The authors in [15] also propose the incorporation of a re-projection step to

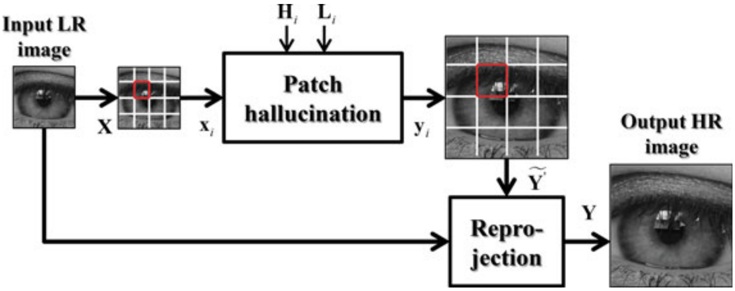


Fig. 5.5 Block diagram of patch-based iris hallucination

reduce artifacts and make the output image more similar to the input image $\mathbf{X}$. The image $\tilde{\mathbf{Y}}'$ is re-projected to $\mathbf{X}$ via gradient descent using

$$\tilde{\mathbf{Y}}^{t+1} = \tilde{\mathbf{Y}}^{t} - \tau \mathbf{U}\left(\mathbf{B}\left(\mathbf{DB}\tilde{\mathbf{Y}}^{t} - \mathbf{X}\right)\right) \tag{5.22}$$

where $\mathbf{U}$ is the up-sampling matrix. The process stops when $|\tilde{\mathbf{Y}}^{t+1} - \tilde{\mathbf{Y}}^{t}| < \varepsilon$.

5.3.2 Local Iterative Neighbour Embedding

Recently, the work [5] adapted the method described in Sect. 5.2.2 to reconstruct iris images, based on multi-layer locality-constrained iterative neighbour embedding (LINE) of local image patches [42]. In the mentioned work [5], and in the experiments reported here, update of the intermediate dictionary has not been implemented. On the other hand, inspired by [15], the re-projection step described in Sect. 5.3.1 has been incorporated in the reconstruction of iris images after the application of the LINE algorithm.

5.3.3 Results

We use the visible spectrum smartphone iris (VSSIRIS) database [66], which consists of images from 28 semi-cooperative subjects (56 eyes) captured using the rear camera of two different smartphones (Apple iPhone 5 S and Nokia Lumia 1020). Images have been obtained in unconstrained conditions under mixed illumination consisting of natural sunlight and artificial room light. Each eye has five samples per smartphone, thus totalling 5 × 56 = 280 images per device (560 in total). Figure 5.6 shows some example images. All images are resized via bicubic interpolation to have the same iris radius using MATLAB's *imresize* function (we choose as target radius the average

Fig. 5.6 Sample images from VSSIRIS database [66]

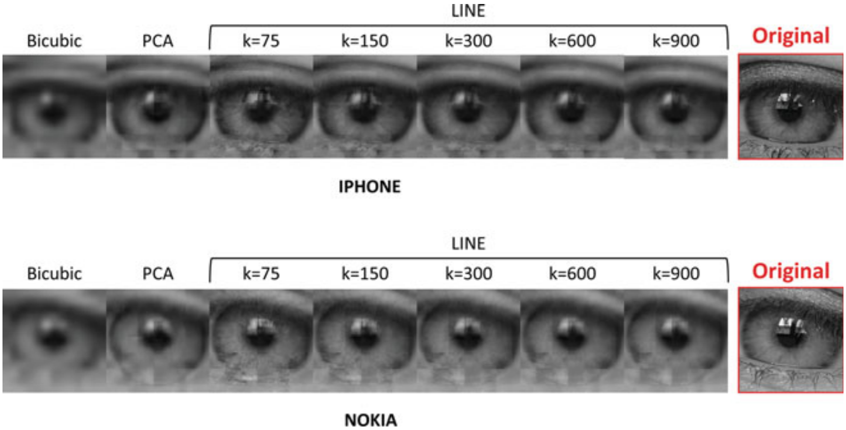


Fig. 5.7 Super-resolved iris images using different iris super-resolution techniques with a magnification factor of ×22

iris radius $R = 145$ of the whole database, given by available ground truth). Then, images are aligned by extracting a square region of 319×319 around the sclera centre (about $1.1 \times R$). Two sample iris images after this procedure can be seen in Fig. 5.7, right.

Aligned and normalized high-resolution images are then down-sampled via bicubic interpolation to different sizes, and then used as input low-resolution images of the reconstruction methods. The low-resolution images are then hallucinated to the original input size. Given an input low-resolution image, we use all available images from the remaining eyes (of both smartphones) to train the hallucination methods (leave-one-out). Training images are mirrored in the horizontal direction to duplicate the size of the training dataset, thus having 55 eyes $\times$ 10 samples $\times$ 2 = 1100 images for training. Verification experiments are done separately for each device. Each eye is considered as a different enrolled user. As enrolment samples, we employ original high-resolution images, whereas reconstructed images are employed as query data. Genuine trials are done by pair-wise comparison between all available images of the same eye, avoiding symmetric matches. Impostor trials are done by comparing the first image of an eye to the second image of the remaining eyes. This procedure results in 56 × 10 = 560 genuine and 56 × 55 = 3018 impostor scores per device.

The results in Tables 5.3 and 5.4 show the performance of different iris super-resolution techniques mentioned in this chapter in terms of both quality (PSNR and SSIM) and equal error rate verification performance, respectively. It can be observed

Table 5.3 Summary of the quality analysis results using the PSNR and SSIM metrics on the VSSIRIS dataset

SR method	Magnification factor									
	2		4		8		16		22	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bicubic	39.2	**0.96**	**33.55**	**0.88**	29.93	**0.8**	26.77	**0.75**	24.97	0.72
Eigen-patches [4]	**39.4**	**0.96**	**33.53**	0.87	**30.19**	0.79	27.37	**0.75**	26	**0.73**
LINE [5] ($k = 75$)	38.43	0.95	31.21	0.78	26.19	0.57	26.13	0.66	25.46	0.68
LINE [5] ($k = 150$)	38.1	0.94	30.24	0.74	27.09	0.62	26.87	0.71	25.85	0.71
LINE [5] ($k = 300$)	37.65	0.93	28.87	0.67	28.76	0.72	27.25	0.73	26.06	0.72
LINE [5] ($k = 600$)	36.8	0.92	30.36	0.74	29.57	0.76	27.41	0.74	**26.15**	**0.73**
LINE [5] ($k = 900$)	35.92	0.9	31.59	0.79	29.82	0.77	**27.45**	0.74	**26.17**	**0.73**

The best result of each column is marked in bold

Table 5.4 Summary of the EER recognition performance (in %) using the Log-Gabor iris recognition algorithm [52] on the VSSIRIS dataset

SR method	iPhone					Nokia				
	Magnification factor					Magnification factor				
	2	4	8	16	22	2	4	8	16	22
High-resolution	8.04					7.5				
Bicubic	14.47	13.88	14.79	16.14	18.47	11.24	10.38	10.88	12.36	14.93
Eigen-patches [4]	8.38	8.33	**7.96**	**8.96**	10.68	**7.61**	**7.09**	**7.3**	9.55	10.54
LINE [5] ($k = 75$)	**7.94**	8.28	8.56	9.98	13.55	7.67	8	8.05	**9.27**	12.65
LINE [5] ($k = 150$)	8.17	8.73	8.12	9.59	12.55	7.75	8.03	8.19	9.81	11.79
LINE [5] ($k = 300$)	8.17	8.52	8.88	9.57	12	7.65	8.56	7.81	9.44	10.75
LINE [5] ($k = 600$)	8.03	8.77	8.54	9.59	11.53	7.65	7.95	7.62	9.98	**10.22**
LINE [5] ($k = 900$)	8.03	**8.11**	8.33	9.61	**10.59**	7.75	7.71	7.64	9.37	10.7

The best result of each column is marked in bold that the two trained reconstruction methods evaluated outperform bicubic interpolation. Its advantage is more evident at very high magnification factors, where the biggest differences in quality metrics and verification performance occur. An interesting observation is that the different evaluation metrics employed here do not show the same tendency or relative difference as resolution changes. For example, the PSNR of bicubic interpolation is similar to that of the best-trained method up to a magnification factor of 8; but with bigger magnification factors, the trained reconstruction methods achieve higher PSNR. The SSIM, on the other hand, remains similar. And despite the PSNR or SSIM being similar or not, the verification performance of bicubic is much worse than the trained methods, regardless of the magnification factor employed. This demonstrates again that image quality metrics are not necessarily good predictors of the recognition performance [6].

Regarding the two trained methods evaluated, there is no clear winner. Regarding the neighbourhood size k of LINE, there are no conclusive results either. The choice of k does not seem to have a significant impact on the performance. Only with a magnification factor of 22, there is a clear tendency for a bigger value of k. It can be seen in Fig. 5.7 that the images restored with a bigger k are more blurred (due to more patches being averaged), but this seems to be positive for the recognition performance nevertheless. It is also worth noting that the verification performance using trained reconstruction remains similar to the baseline performance up to a magnification factor of 8 (which corresponds to an image size of only 29×29). This would allow to keep query images of very low size without sacrificing performance, with positive implications, for example, if there are data storage or transmission restrictions.

5.4 Summary and Future Trends

Face and iris biometrics are two well-explored modalities, with systems yielding state-of-the-art performance in controlled scenarios. However, the use of more relaxed acquisition environments, like the one represented in selfie biometrics, is pushing image-based biometrics towards the use of low-resolution imagery. If not tackled properly, low-resolution images can pose significant problems in terms of reduced performance. In this context, super-resolution techniques can be used to enhance the quality of low-resolution images to improve the recognition performance of existing biometric systems.

Super-resolution is a core topic in computer vision, with many techniques proposed to restore low-resolution images [54, 73]. However, compared with the existing literature in generic super-resolution, super-resolution in biometrics is a relatively recent topic [60]. This is because most approaches are general scene, designed to produce overall visual enhancement. They try to improve the quality of the image by minimizing objective measures, such as the peak signal-to-noise (PSNR), which does not necessarily correlate with better recognition performance [3]. Images from a specific biometric modality have particular local and global structures that can be exploited to achieve a more efficient up-sampling [8]. For example, recovering local texture details is essential for face and ocular images due to the prevalence of texture-based recognition in these modalities [40]. This chapter has presented an overview of the image super-resolution topic, with emphasis on the reconstruction of face and iris images in the visible spectrum, which are the two prevalent modalities in selfie biometrics. We investigate several existing techniques and evaluate their application to reconstruct face and iris images.

Despite promising performance of super-resolution methods for facial or ocular images under well-controlled conditions, they degenerate when encountering images from uncontrolled environments, as, for example, non-frontal views, expression or lightning changes [76]. Future trends in biometrics super-resolution therefore relate to designing effective approaches to cope with these variations. For example, one limitation of existing studies is that low-resolution images are simulated by down-sampling high-resolution images due to the lack of databases with low-resolution and corresponding high-resolution reference images. As a result, variations in pose, illumination or expression are not yet fully considered, neither the associated artifacts introduced (e.g. compression, noise or blur). In addition, prior to down-sampling, images are aligned by manual annotation of landmarks (eyes, nose, etc.) followed by affine transformation. All super-resolution schemes employed in the biometric literature are heavily affected by imprecise image alignment, even by small amounts; however, real low-resolution images usually have blurring, and so many ambiguities exist for landmark localization or segmentation, thus making a necessity the use of reconstruction schemes capable of working under imprecise alignment.

Acknowledgements Authors F. A.-F. and J. B thank the Swedish Research Council (VR), the Sweden's innovation agency (VINNOVA), and the Swedish Knowledge Foundation (CAISR programme and SIDUS-AIR project). Author J. F. thanks Accenture and the project CogniMetrics (TEC2015-70627-R) from MINECO/FEDER

References

1. Ahonen T, Hadid A, Pietikainen M (2006) Face description with local binary patterns: application to face recognition. IEEE Trans Pattern Anal Mach Intell 28(12):2037–2041. https://doi.org/10.1109/TPAMI.2006.244
2. Aljadaany R, Luu K, Venugopalan S, Savvides M (2015) Iris super-resolution via nonparametric over-complete dictionary learning. In: Proceedings of the IEEE international conference on image processing, ICIP, pp 3856–3860. https://doi.org/10.1109/ICIP.2015.7351527
3. Alonso-Fernandez F, Bigun J (2013) Quality factors affecting iris segmentation and matching. In: Proceedings of the international conference on biometrics, ICB, pp 1–6. https://doi.org/10.1109/ICB.2013.6613016
4. Alonso-Fernandez F, Farrugia RA, Bigun J (2015) Eigen-patch iris super-resolution for iris recognition improvement. In: Proceedings of the European signal processing conference, EUSIPCO
5. Alonso-Fernandez F, Farrugia RA, Bigun J (2017) Iris super-resolution using iterative neighbor embedding. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, CVPRW, pp 655–663. https://doi.org/10.1109/CVPRW.2017.94

6. Alonso-Fernandez F, Fierrez J, Ortega-Garcia J (2012) Quality measures in biometric systems. IEEE Secur Privacy 10(6):52–62
7. Baker S, Kanade T (2000) Hallucinating faces. In: Proceedings fourth IEEE international conference on automatic face and gesture recognition (Cat. No. PR00580), pp 83–88 . https://doi.org/10.1109/AFGR.2000.840616
8. Baker S, Kanade T (2002) Limits on super-resolution and how to break them. IEEE Trans Pattern Anal Mach Intell 24(9):1167–1183
9. Barnard R, Pauca VP, Torgersen TC, Plemmons RJ, Prasad S, van der Gracht J, Nagy J, Chung J, Behrmann G, Mathews S, Mirotznik M (2006) High-resolution iris image reconstruction from low-resolution imagery. https://doi.org/10.1117/12.681930
10. Cao Q, Lin L, Shi Y, Liang X, Li G (2017) Attention-aware face hallucination via deep reinforcement learning. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR), pp 1656–1664. https://doi.org/10.1109/CVPR.2017.180
11. Carrato S, Ramponi G, Marsi S (1996) A simple edge-sensitive image interpolation filter. In: Proceedings of 3rd IEEE international conference on image processing, vol 3, pp 711–714
12. Chakrabarti A, Rajagopalan AN, Chellappa R (2007) Super-resolution of face images using kernel pca-based prior. IEEE Trans Multimedia 9(4):888–892. https://doi.org/10.1109/TMM.2007.893346
13. Chang H, Yeung DY, Xiong Y (2004) Super-resolution through neighbor embedding. In: Proceedings of the 2004 IEEE computer society conference on computer vision and pattern recognition, 2004. CVPR 2004, vol 1, p I
14. Cheeseman P, Kanefsky B, Kraft R, Stutz J, Hanson R (1996) Super-resolved surface reconstruction from multiple images. Springer Netherlands, Dordrecht, pp 293–308
15. Chen HY, Chien SY (2014) Eigen-patch: position-patch based face hallucination using eigen transformation. In: Proceedings of the IEEE international conference on multimedia and expo, ICME, pp 1–6
16. Deshpande A, Patavardhan P (2017) Multi-frame super-resolution for long range captured iris polar image. IET Biomet 6(2):108–116. https://doi.org/10.1049/iet-bmt.2016.0076
17. Deshpande A, Patavardhan PP (2017) Super resolution and recognition of long range captured multi-frame iris images. IET Biomet 6(5):360–368. https://doi.org/10.1049/iet-bmt.2016.0075
18. Dong C, Loy CC, He K, Tang X (2016) Image super-resolution using deep convolutional networks. IEEE Trans Pattern Anal Mach Intell 38(2):295–307. https://doi.org/10.1109/TPAMI.2015.2439281
19. Dong W, Zhang L, Shi G, Li X (2013) Nonlocally centralized sparse representation for image restoration. IEEE Trans Image Process 22(4):1620–1630. https://doi.org/10.1109/TIP.2012.2235847
20. Fahmy G (2007) Super-resolution construction of iris images from a visual low resolution face video. In: Proceedigns of the national radio science conference, NRSC
21. Farrugia RA, Guillemot C (2016) Robust face hallucination using quantization-adaptive dictionaries. In: 2016 IEEE international conference on image processing (ICIP), pp 414–418 . https://doi.org/10.1109/ICIP.2016.7532390
22. Farrugia RA, Guillemot C (2017) Face hallucination using linear models of coupled sparse support. IEEE Trans Image Process 26(9):4562–4577. https://doi.org/10.1109/TIP.2017.2717181
23. Farsiu S, Robinson D, Elad M, Milanfar P (2003) Fast and robust super-resolution. In: Proceedings 2003 international conference on image processing (Cat. No.03CH37429), vols 2 and 3, pp II–291–4. https://doi.org/10.1109/ICIP.2003.1246674
24. Farsiu S, Robinson D, Elad M, Milanfar P (2003) Robust shift and add approach to superresolution. https://doi.org/10.1117/12.507194
25. Farsiu S, Robinson MD, Elad M, Milanfar P (2004) Fast and robust multiframe super resolution. IEEE Trans Image Process 13(10):1327–1344. https://doi.org/10.1109/TIP.2004.834669
26. Fierrez J, Morales A, Vera-Rodriguez R, Camacho D (2018) Multiple classifiers in biometrics. Part 1: fundamentals and review. Inf Fusion 44:57–64 . https://doi.org/10.1016/j.inffus.2017.12.003

27. Fierrez J, Morales A, Vera-Rodriguez R, Camacho D (2018) Multiple classifiers in biometrics. Part 2: trends and challenges. Inf Fusion 44:103–112 . https://doi.org/10.1016/j.inffus.2017.12.005
28. Freeman WT, Jones TR, Pasztor EC (2002) Example-based super-resolution. IEEE Comput Graph Appl 22(2):56–65. https://doi.org/10.1109/38.988747
29. Galbally J, Marcel S, Fierrez J (2014) Image quality assessment for fake biometric detection: application to iris, fingerprint, and face recognition. IEEE Trans Image Process 23(2):710–724
30. Glasner D, Bagon S, Irani M (2009) Super-resolution from a single image. In: The IEEE international conference on computer vision (ICCV)
31. Gonzalez RC, Woods RE (2006) Digital image processing, 3rd edn. Prentice-Hall Inc., Upper Saddle River, NJ, USA
32. Hardie RC, Barnard KJ, Armstrong EE (1997) Joint map registration and high-resolution image estimation using a sequence of undersampled images. IEEE Trans Image Process 6(12):1621–1633. https://doi.org/10.1109/83.650116
33. Hollingsworth K, Peters T, Bowyer K, Flynn P (2009) Iris recognition using signal-level fusion of frames from video. IEEE Trans Inf Forensic Secur 4(4):837–848
34. Hsieh SH, Li YH, Tien CH, Chang CC (2016) Extending the capture volume of an iris recognition system using wavefront coding and super-resolution. IEEE Trans Cybern 46(12):3342–3350. https://doi.org/10.1109/TCYB.2015.2504388
35. Hu Y, Lam KM, Qiu G, Shen T (2011) From local pixel structure to global image super-resolution: a new face hallucination framework. IEEE Trans Image Process 20(2):433–445. https://doi.org/10.1109/TIP.2010.2063437
36. Huang H, He H, Fan X, Zhang J (2010) Super-resolution of human face image using canonical correlation analysis. Pattern Recogn 43(7):2532–2543. https://doi.org/10.1016/j.patcog.2010.02.007
37. Huang J, Ma L, Tan T, Wang Y (2003) Learning based resolution enhancement of iris images. In: Proceedings of the BMVC
38. Irani M, Peleg S (1990) Super resolution from image sequences. In: [1990] Proceedings. 10th international conference on pattern recognition, vol 2, pp 115–120. https://doi.org/10.1109/ICPR.1990.119340
39. Irani M, Peleg S (1993) Motion analysis for image enhancement: resolution, occlusion, and transparency. J Vis Commun Image Represent 4(4):324–335. https://doi.org/10.1006/jvci.1993.1030. URL http://www.sciencedirect.com/science/article/pii/S1047320383710308
40. Jain A, Nandakumar K, Ross A (2016) 50 years of biometric research: accomplishments, challenges, and opportunities. Pattern Recogn Lett 79:80–105
41. Jain AK, Kumar A (2011) Second generation biometrics, chapter. An overview. Springer, Biometrics of next generation
42. Jiang J, Hu R, Wang Z, Han Z (2014) Face super-resolution via multilayer locality-constrained iterative neighbor embedding and intermediate dictionary learning. IEEE Trans Image Proces 23(10):4220–4231. https://doi.org/10.1109/TIP.2014.2347201
43. Jillela R, Ross A, Flynn P (2011) Information fusion in low-resolution iris videos using principal components transform. In: Proceedings of the IEEEwWorkshop on applications of computer vision, WACV, pp 262–269. https://doi.org/10.1109/WACV.2011.5711512
44. Jung C, Jiao L, Liu B, Gong M (2011) Position-patch based face hallucination using convex optimization. IEEE Sig Process Lett 18(6):367–370. https://doi.org/10.1109/LSP.2011.2140370
45. Kim J, Kwon Lee J, Mu Lee K (2016) Accurate image super-resolution using very deep convolutional networks. In: The IEEE conference on computer vision and pattern recognition (CVPR)
46. Li B, Chang H, Shan S, Chen X (2009) Aligning coupled manifolds for face hallucination. IEEE Sig Process Lett 16(11):957–960. https://doi.org/10.1109/LSP.2009.2027657
47. Li Y, Cai C, Qiu G, Lam KM (2014) Face hallucination based on sparse local-pixel structure. Pattern Recogn 47(3), 1261–1270. https://doi.org/10.1016/j.patcog.2013.09.012. URL http://www.sciencedirect.com/science/article/pii/S0031320313003841 (Handwriting recognition and other PR applications)

48. Lim B, Son S, Kim H, Nah S, Lee KM (2017) Enhanced deep residual networks for single image super-resolution. In: 2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW), pp 1132–1140. https://doi.org/10.1109/CVPRW.2017.151
49. Lin Z, Shum HY (2004) Fundamental limits of reconstruction-based superresolution algorithms under local translation. IEEE Trans Pattern Anal Mach Intell 26(1):83–97
50. Liu J, Sun Z, Tan T (2013) Code-level information fusion of low-resolution iris image sequences for personal identification at a distance. In: Proceedings of the international conference on biometrics: theory, applications and systems, BTAS, pp 1–6. https://doi.org/10.1109/BTAS.2013.6712692
51. Ma X, Zhang J, Qi C (2009) Position-based face hallucination method. In: 2009 IEEE international conference on multimedia and expo, pp 290–293. https://doi.org/10.1109/ICME.2009.5202492
52. Masek L (2003) Recognition of human iris patterns for biometric identification. Master's thesis. School of Computer Science and Software Engineering, University of Western Australia
53. Nasir H, Stankovic V, Marshall S (2011) Singular value decomposition based fusion for super-resolution image reconstruction. In: 2011 IEEE international conference on signal and image processing applications (ICSIPA), pp 393–398. https://doi.org/10.1109/ICSIPA.2011.6144138
54. Nasrollahi K, Moeslund TB (2014) Super-resolution: a comprehensive survey. Mach Vis Appl 25(6):1423–1468. https://doi.org/10.1007/s00138-014-0623-4
55. Nguyen K, Fookes C, Sridharan S (2010) Robust mean super-resolution for less cooperative nir iris recognition at a distance and on the move. In: Proceedings of the symposium on information and communication technology, SoICT, pp 122–127 . https://doi.org/10.1145/1852611.1852635
56. Nguyen K, Fookes C, Sridharan S., Denman S (2010) Focus-score weighted super-resolution for uncooperative iris recognition at a distance and on the move. In: Proceedings of the 25th international conference of image and vision computing New Zealand, IVCNZ, pp 1–8. https://doi.org/10.1109/IVCNZ.2010.6148792
57. Nguyen K, Fookes C, Sridharan S, Denman S (2011) Feature-domain super-resolution for iris recognition. In: Proceedings of the IEEE international conference on image processing, ICIP, pp 3197–3200. https://doi.org/10.1109/ICIP.2011.6116348
58. Nguyen K, Fookes C, Sridharan S, Denman S (2011) Quality-driven super-resolution for less constrained iris recognition at a distance and on the move. IEEE Trans Inf For Secur 6(4):1248–1258
59. Nguyen K, Fookes C, Sridharan S, Denman S (2013) Feature-domain super-resolution for iris recognition. Comput Vis Image Underst 117(10):1526–1535. https://doi.org/10.1016/j.cviu.2013.06.010. URL http://www.sciencedirect.com/science/article/pii/S1077314213001306
60. Nguyen K, Fookes C, Sridharan S, Tistarelli M, Nixon M (2018) Super-resolution for biometrics: a comprehensive survey. Pattern Recogn 78:23–42. https://doi.org/10.1016/j.patcog.2018.01.002. URL http://www.sciencedirect.com/science/article/pii/S0031320318300049
61. Nguyen K, Sridharan S, Denman S, Fookes C (2012) Feature-domain super-resolution framework for gabor-based face and iris recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, CVPR, pp 2642–2649
62. Othman N, Dorizzi B (2015) Impact of quality-based fusion techniques for video-based iris recognition at a distance. IEEE Trans Inf For Secur 10(8):1590–1602. https://doi.org/10.1109/TIFS.2015.2421314
63. Park JS, Lee SW (2008) An example-based face hallucination method for single-frame, low-resolution facial images. IEEE Trans Image Process 17(10):1806–1816. https://doi.org/10.1109/TIP.2008.2001394
64. Parkhi OM, Vedaldi A, Zisserman A (2015) Deep face recognition. In: Proceedings of the British machine vision conference, BMVC
65. Pham TQ, van Vliet LJ, Schutte K (2006) Robust fusion of irregularly sampled data using adaptive normalized convolution. EURASIP J Adv Signal Process 2006(1):083,268. https://doi.org/10.1155/ASP/2006/83268

66. Raja KB, Raghavendra R, Vemuri VK, Busch C (2015) Smartphone based visible iris recognition using deep sparse filtering. Pattern Recogn Lett 57:33–42
67. Ribeiro E, Uhl A, Alonso-Fernandez F, Farrugia RA (2017) Exploring deep learning image super-resolution for iris recognition. In: Proceedings of the 25th European signal processing conference, EUSIPCO, pp 2176–2180. https://doi.org/10.23919/EUSIPCO.2017.8081595
68. Schultz RR, Stevenson RL (1996) Extraction of high-resolution frames from video sequences. IEEE Trans Image Process 5(6):996–1011. https://doi.org/10.1109/83.503915
69. Shin KY, Park KR, Kang BJ, Park SJ (2009) Super-resolution method based on multiple multi-layer perceptrons for iris recognition. In: International conference ubiquitous information technologies applications, ICUT, pp 1–5
70. Simonyan K, Grishin S, Vatolin D, Popov D (2008) Fast video super-resolution via classification. In: 2008 15th IEEE international conference on image processing, pp 349–352. https://doi.org/10.1109/ICIP.2008.4711763
71. Su D, Willis P (2004) Image interpolation by pixel-level data-dependent triangulation. Comput Graph Forum. https://doi.org/10.1111/j.1467-8659.2004.00752.x
72. Su K, Tian Q, Xue Q, Sebe N, Ma J (2005) Neighborhood issue in single-frame image super-resolution. In: 2005 IEEE international conference on multimedia and expo, p 4. https://doi.org/10.1109/ICME.2005.1521623
73. Thapa D, Raahemifar K, Bobier WR, Lakshminarayanan V (2016) A performance comparison among different super-resolution techniques. Comput Electr Eng 54:313–329. https://doi.org/10.1016/j.compeleceng.2015.09.011. URL http://www.sciencedirect.com/science/article/pii/S0045790615003183
74. Thévenaz P, Blu T, Unser M (2000) Handbook of medical imaging. Chapter. Image interpolation and resampling. Academic Press, Inc., Orlando, pp 393–420. URL http://dl.acm.org/citation.cfm?id=374166.374424
75. Turk M, Pentland A (1991) Eigenfaces for recognition. J Cogn Neurosci 3(1):71–86
76. Wang N, Tao D, Gao X, Li X, Li J (2014) A comprehensive survey to face hallucination. Int J Comput Vis 106(1):9–30
77. Wang X, Tang X (2005) Hallucinating face by eigentransformation. IEEE Trans Syst Man Cybern Part C Appl Rev 35(3):425–434. https://doi.org/10.1109/TSMCC.2005.848171
78. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 13(4):600–612. https://doi.org/10.1109/TIP.2003.819861
79. Yang J, Wright J, Huang T, Ma Y (2008) Image super-resolution as sparse representation of raw image patches. In: 2008 IEEE conference on computer vision and pattern recognition, pp 1–8. https://doi.org/10.1109/CVPR.2008.4587647
80. Yang J, Wright J, Huang TS, Ma Y (2010) Image super-resolution via sparse representation. IEEE Trans Image Process 19(11):2861–2873. https://doi.org/10.1109/TIP.2010.2050625
81. Zhang Q, Li H, He Z, Sun Z (2016) Image super-resolution for mobile iris recognition. In: Proceedings of the 11th Chinese conference on biometric recognition, CCBR, pp 399–406
82. Zomet A, Peleg S (2000) Efficient super-resolution and applications to mosaics. In: Proceedings 15th international conference on pattern recognition. ICPR-2000, vol 1, pp 579–583. https://doi.org/10.1109/ICPR.2000.905404

Chapter 6
Foveated Vision for Biologically Inspired Continuous Face Authentication

Souad Khellat-Kihel, Andrea Lagorio and Massimo Tistarelli

Abstract In everyday life whenever people observe, interact or speak to each other, visual attention is mostly directed toward the other person's face, particularly to the eyes and the nearby periocular regions. This is naturally reflected when the user interacts with their mobile phones in several usual activities, such as web access, payments and video calls. For this reason, the functionality of mobile devices is strongly affected by the design of the user interface. In this chapter, we propose a biologically inspired approach for continuous user authentication based on the analysis of the ocular regions. The proposed system is based on a modified version of the HMAX visual processing module. HMAX is a hierarchical model which has been conceived to mimic the basic neural architecture of the ventral stream of the visual cortex. The original HMAX model consists of four layers: S1, C1, S2 and C2. S1 and C1 represent the responses to a bank of orientation-selective Gabor filters. S2 and C2 represent the responses of simple and complex cells to other textural features. The discrimination power of HMAX in recognizing classes of objects is invariant to rotation and scale. The C1 layer, which is mainly responsible for the scale and rotation invariance, is implemented using a max-pooling operation, which may lose some spatial information. To overcome this problem while preserving the maximal visual acuity and hence the localization accuracy, we propose to augment the model by applying a retinal log-polar mapping. The log-polar mapping is an approximation of the retino-cortical mapping that is performed by the early stages of the primate visual system. Due to the high density of the cones in the fovea, the log-polar approximation of the space-variant distribution model of the photoreceptors can only be applied outside the foveal region. Therefore, the log-polar mapping is added to the HMAX model as a complementary stage to process the peripheral region of the grabbed images. In order to demonstrate the feasibility of the proposed approach to mobile scenarios, experimental results obtained from publicly available databases and image streams grabbed from mobile devices will be presented.

S. Khellat-Kihel · A. Lagorio · M. Tistarelli (✉)
Computer Vision Laboratory, University of Sassari, Sassari, Italy
e-mail: tista@uniss.it

S. Khellat-Kihel
e-mail: skhellatkihel@uniss.it

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_6

Keywords Face recognition · HMAX architecture · Log-polar mapping · Visual perception · Retinal vision

6.1 Introduction

Object recognition, in general, and face recognition, in particular, can be performed either in a holistic manner, by processing the entire image, or as a local process, by analyzing a series of local regions of the image containing the object or the face to be recognized. Concerning face recognition, for un-cooperative or loosely cooperative scenarios, part-based approaches have the advantage of coping up better with occlusions, make-up or other adversarial conditions [1, 2]. This chapter proposes a part-based approach to face recognition, for un-cooperative scenarios such as continuous authentication on mobile devices. The core of the algorithm is based on the HMAX network, which is an approximate model of the early stages of the visual pathway in primates. Even though the HMAX model well mimics the anatomical architecture of the primary visual area (V1), it does not take into account the retino-cortical mapping which takes advantage of the space-variant topology of the human retina. The proposed approach aims to fill this gap by adding a retino-cortical transformation which has an advantage to the scale- and rotation-invariant properties of the retinal topology.

The original HMAX model consists of four layers (S1, C1, S2 and C2) each layer mimicking the responses of either simple cells, complex cells or hyper complex cells. S1 and C1 can also be simulated with the responses of a bank of Gabor filters with several orientations and scales, while S2 and C2 can be modeled with the responses of more complex filters.

The logarithmic to polar (or log-polar) mapping was originally proposed to model the space-variant arrangement of the ganglion cells in the human retina [3]. In the proposed approach, this approach is adopted as a preprocessing stage to implement the retino-cortical transformation.

As noted in [4], the image areas lying inside the face are not the best suited to perform unsupervised learning with the HMAX architecture. Therefore, the developed computational framework includes the processing of the image area around the face. The outer face is represented by means of the log-polar transformation [3].

The final classification stage is implemented by means of a softmax as an activation layer, while cross-entropy loss is the loss function adopted for error estimation.

6.2 Related Work

The most recent approaches for face recognition on mobile devices are based on active and/or continuous authentication [5]. In [4], local regions (left and right eye, nose and mouth) are extracted from a set of fiducial points. The extracted textured regions are concatenated to obtain a vector for classification. Weng et al. [2] presented an algorithm for face detection based on convolutional deep networks. This algorithm has been proposed to be suitable for mobile devices. In [3], continuous authentication is performed by matching facial attributes. The user authentication is performed by simply comparing the computed facial attributes with the enrolled attributes of the original user. Several algorithms for face recognition on mobile devices have been proposed and compared in [6]. These methods are based on the comparison of intensity values, LBP features and applying transfer learning from the five layers of AlexNet. Two additional methods are based on the direct comparison of fiducial points. The best performance is obtained with the DCNN and the cosine distance obtaining an EER which is less than 5%.

The most recent methods for face recognition are based on the deep convolutional neural network [6–8]. Even though deep neural networks have been successfully used to address several hard problems in computer vision, the implementation is computationally intensive and the network is generally designed with each layer processing the entire image. The HMAX model, which was developed before CNNs started to be extensively used to solve computer vision problems, was developed to demonstrate the feasibility of a biologically plausible architecture for face recognition. The model was tested on several publicly available databases such as LFW, PubFig and SURF-W [9] providing results comparable to state-of-the-art methods.

In [10], a new C3EFs inspired from the ventral and dorsal stream of visual cortex have been used. They proposed a model to extract new view-independent features, using visual attention model and ventral stream model to achieve the goal of view-independent face recognition. A higher layer has been added to the HMAX model. Hu et al. [11] proposed an improved version of the HMAX model, named as sparse HMAX. This model addresses the local-to-global structure gradually along the hierarchy by applying a patch-based learning approach to the output of the previous layer. The major difference between the two models is that in the sparse HMAX S2 bases are learned by sparse coding, and therefore the S2 codes are computed by sparse coding.

6.3 Face Recognition on Mobile Devices

Capturing and processing face images on mobile devices are generally an easy process as long as the user is cooperative. However, if the face acquisition and recognition process are continuous and without the explicit cooperation of the user, most of the grabbed face images can be partially occluded, with strongly uneven illumination and unpredictable motion and pose. Processing substantially degraded images require taking advantage of multiple sources of information and using piecewise polynomial for missing data.

When analyzing the outputs of the inner layers of several deep convolutional neural networks, it turns out that the data processed at each layer is related to the area inside the face and those around it. This is in strong contrast with the approach adopted by almost all conventional face recognition algorithms that were based on carefully cropping of the inner portion of the faces, in order to exclude outer part of the face. The remarkable performance produced by the most recent face recognition CNN architectures already demonstrates the high discrimination power of the outer face area. For this reason, in this chapter, a modified version of the HMAX hierarchical model is proposed. The proposed method includes both a foveal visual process, dedicated to analyzing the inner facial regions, and a peripheral visual process, dedicated to analyzing the information which can be captured from the regions around the face. The proposed approach is based on a multistage architecture involving the detection of the face, the extraction of 64 standard facial landmarks and the ocular regions. A log-polar transformation is independently applied to the extracted ocular regions and the entire face image. The log-polar parametrization allows to extract the information on the ocular regions with high resolution thus mimicking the retinal foveal vision, while the entire face is re-sampled to retain only the peripheral portion of the face. The HMAX architecture is fed with both the foveal and the peripheral data for feature extraction. Finally, a softmax layer is applied for classification. The general structure of the proposed framework is depicted in Fig. 6.1.

6.3.1 *Detection of the Face, Ocular Regions and the Landmarks*

Faces in the image are detected, and the eyes and mouth regions are extracted by applying the Viola–Jones algorithm [12]. The Viola–Jones algorithm may fail in some cases, such as if the face is darker than the background, as it happens with a strong backlight. In other cases, the algorithm is unable to correctly classify the eyes or other facial regions. Some examples of mis-classification are depicted in Fig. 6.2.

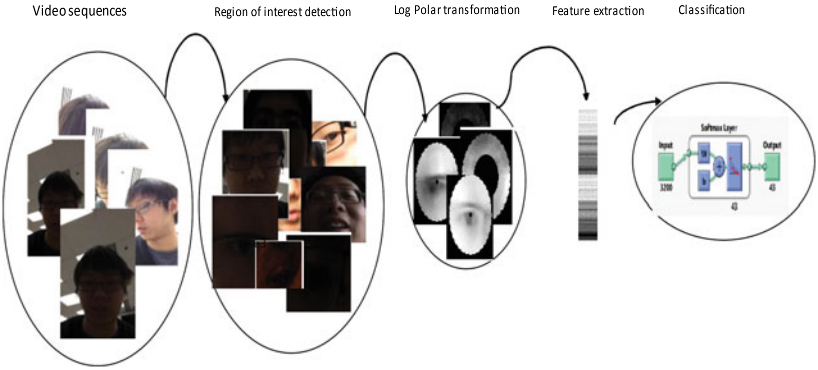


Fig. 6.1 General structure of the proposed framework

In order to improve the face detection process, the landmark extraction algorithm is applied to detect the face[1] and the facial landmarks. The adopted algorithm models each facial landmark as a local region. A global optimization is then applied to capture topological modifications due to changes in viewpoint or head pose [13]. The landmarks are also used to select the images out of a sequence where the face is best viewed. The local structure of the landmarks within the ocular regions is analyzed to determine if the eyes are open, and consequently the face can be reliably detected and used for further processing. For both eyes, the arctangent of the angle between the top, bottom and the outer landmark points is computed. If the average arctangent value computed from the eyes is less than 1, then either one or both the eyes are open. Therefore, the frame is selected, and the facial and ocular regions are extracted (Figs. 6.3 and 6.4).

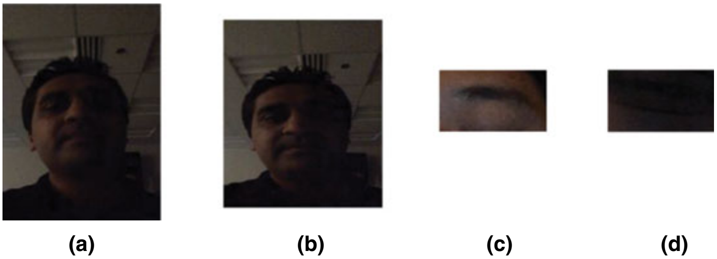
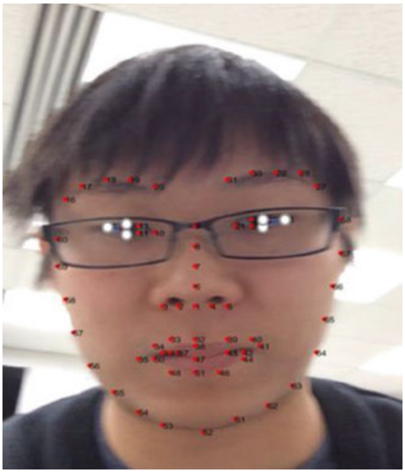


Fig. 6.2 Errors recorded performing the face and eyes detection with the Viola–Jones algorithm. **a** and **b** The face is not detected because it is darker than the background. **c** The eyebrow is mistakenly extracted as the ocular region. **d** Incorrect detection of the region of interest

Fig. 6.3 Sample image from the UMD database with the extracted landmark points



[1] In this case, the landmark extraction algorithm is applied as an alternative method to the Viola–Jones algorithm for the detection of the face and the extraction of the regional regions.

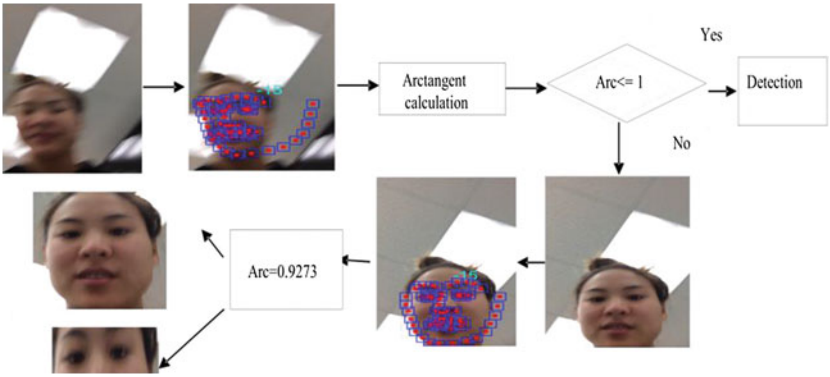


Fig. 6.4 Illustrates the process of facial region selection based on the arctangent test on the landmark points around the eyes

6.3.2 *Foveal and Peripheral Vision*

Looking at a face from a short distance should produce a different perception of the same face viewed from a long distance. However, the human visual system is capable of coping up with the size change due to distance by capturing a high-resolution description of the most salient features of the viewed face. In an artificial system, this can be accomplished either by "foveating," in rapid succession, these parts of the scene or moving an interest window on a high-resolution image [14]. Certainly, facial features are important for recognition, but it cannot be said that the face itself is better characterized by the most prominent features taken in isolation, rather than by the context in which they are located. For this reason, it is not sufficient to scan the face or the image with a fovea, but it is also necessary to provide some information on the area around the fovea. A way to meet both requirements is to adopt a space-variant sampling strategy of the image where the central part of the visual field is sampled at a higher resolution than the periphery. In this way, the peripheral part of the visual field is coded at low resolution but can be still used to describe the context in which foveal information is located. A great advantage of this approach is the considerable data reduction with respect to adopting a uniform resolution schema, while a wide field of view (i.e., peripheral vision) is preserved [15, 16].

The high acuity in the fovea is due to the dense packing of cone photoreceptors (Fig. 6.5). On the other hand, the low acuity in the peripheral area of the retina is due to the lack of cones and the relative sparsity of rods [17]. In the proposed system, the fovea is directed to capture the information lying on the ocular regions, while the periphery captures the information on the outer region of the entire face.

The arrangement of the cones in the human retina, and the corresponding variable size of the ganglion cells receptive fields across the visual field, produces a space-variant topological transformation of the retinal image into its cortical projection

[18]. This transformation can be presented by a log-polar mapping. This retino-cortical mapping can be described through a transformation from the retinal plane onto the cortical plane, which is scale and rotation-invariant, as depicted in Fig. 6.6. If (x, y) are Cartesian coordinates and (ρ, θ) are the polar coordinates, by denoting $z = x + jy = \rho e^{j\theta}$ a point in the complex plane, the log-polar mapping is

$$w = \ln(z). \tag{6.2}$$

As the resolution in the fovea is almost constant, this transformation is a good approximation of the non-foveal part of the retinal image. Therefore, it is applied to reproduce peripheral vision by re-sampling the outer region of the face.

The transformation has been implemented through the algorithm proposed in [19]. Different parameters can be tuned, such as the number of cells per eccentricity (CP), the number of eccentricities (NE), the cell dimension (CD), the size of the overlapping area along the eccentricity (OE) and the radius (OR).

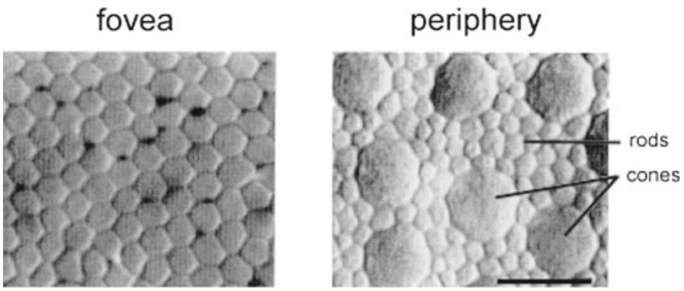


Fig. 6.5 Cones distribution in fovea and periphery

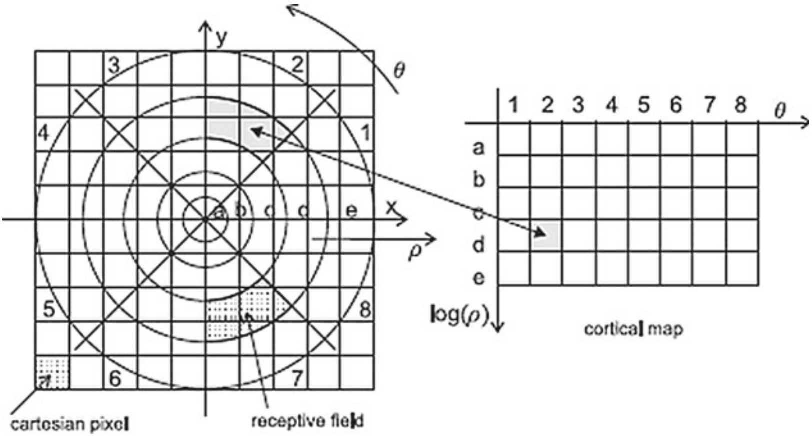


Fig. 6.6 Diagram explaining the log-polar transformation. Every pixel $[x, y]$ on the Cartesian plane is represented on the basis (ρ, θ) as $[\ln \theta]$ on the cortical plane

The Caltech database, composed of 450 frontal face images of 27 subjects [20], is employed to select the parameters for the log-polar mapping. Several classification experiments were carried on the remapped face images, by assigning different values to the parameters of the log-polar transformation. The resulting scores are reported in Table 6.1. In Fig. 6.7, some example images remapped with the obtained best parameters are shown.

Table 6.1 EER and GAR@ 1% FAR of the facial regions to tune the log-polar mapping parameters

Parameters (CP, NE, CD, OE, OR)	Used regions	EER (Equal Error Rate)	VR at 1% FAR (Verification Rate at 1% False Acceptance Rate)
(50, 50, 50, 1.5, 1.5)	Left eye	2	98
(50, 50, 50, 1.5, 1.5)	Right eye	0.22	100
(50, 50, 50, 1.7, 1.7)	Left eye	2	98
(50, 50, 50, 1.7, 1.7)	Right eye	0.44	98
(70, 50, 50, 1.5, 1.5)	Left eye	2	96
(70, 50, 50, 1.5, 1.5)	Right eye	1.67	98
(70, 50, 50, 1.7, 1.7)	Left eye	4	96
(70, 50, 50, 1.7, 1.7)	Right eye	2	98
(32, 32, 120, 1.7, 1.7)	Left eye	0.22	100
(32, 32, 120, 1.7, 1.7)	Right eye	0	100
(32, 64, 130, 1.5, 1.5)	Face	0.11	100
(32, 64, 130, 1.5, 1.5)	Face	0	100

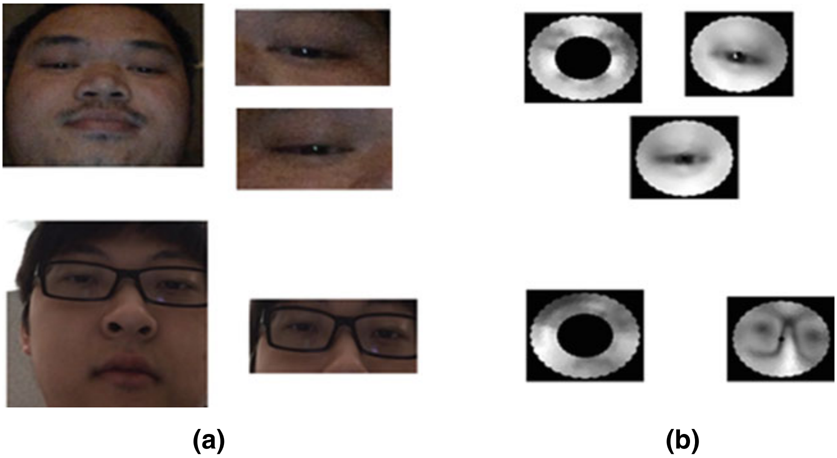


Fig. 6.7 Examples of images remapped according to the log-polar transformation: **a** original images, **b** images mapped on the log-polar plane

6.3.3 The Original HMAX Model

The HMAX model is a hierarchical model for object representation and recognition inspired by the neural architecture of the early stages of the visual cortex in the primates. The general architecture of the HMAX model is represented in Fig. 6.8. Proceeding to the higher levels of the model, the number and typicality of the extracted features change. Each layer is projected to the next layer by applying template matching or max-pooling filters. Proceeding to the higher levels of the model, the number of (*X*, *Y*) pixel positions in a layer is reduced. The input to the model is the gray level image. S1 and C1 represent the responses to a bank of Gabor filters tuned to different orientations. S2 and C2 are the responses to more complex filtering stages.

The first layer S1 in the HMAX network consists of a bank of Gabor filters applied to the full resolution image. The response to a particular filter G, of layer S, at the pixel position (*X*, *Y*) is given by:

$$R(\boldsymbol{X},\boldsymbol{Y}) = \left|\frac{\sum \boldsymbol{X}_i \boldsymbol{G}_i}{\sqrt{\sum \boldsymbol{X}_i^2}}\right| \tag{6.3}$$

The size of the Gabor filter is 11 × 11 ,and it is formulated as follows:

$$\boldsymbol{G}(\boldsymbol{x},\boldsymbol{y}) = \mathbf{exp}\left(-\frac{(\boldsymbol{x}^2+\boldsymbol{\gamma}^2\boldsymbol{Y}^2)}{2\boldsymbol{\sigma}^2}\right)\mathbf{cos}\left(\frac{2\boldsymbol{\pi}}{\boldsymbol{\lambda}}\boldsymbol{X}\right) \tag{6.4}$$

where $X = x\ \cos\theta - y\sin\theta$ and $Y = x\sin\theta + y\cos\theta$. x and y vary between -5 and 5, and θ varies between 0 and π. The parameters γ (aspect ratio), σ (effective width)

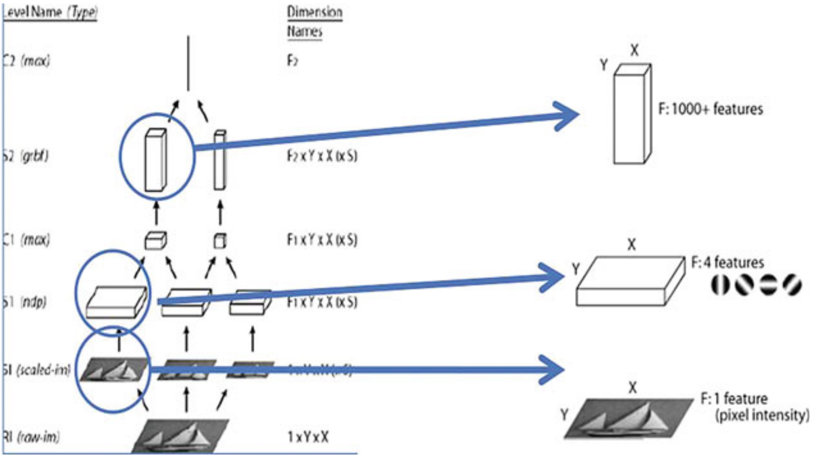


Fig. 6.8 General architecture of the HMAX model

and λ (wavelength) are set to 0.3, 4.5 and 5.6, respectively. For the local invariance (C1) layer, a local maximum is computed for each orientation. They also perform a subsampling by a factor of 5 in both the X and Y directions [20]. In the intermediate feature layer (S2 level), the response for each C1 grid position is computed. Each feature is tuned to a preferred pattern as a stimulus. Starting from an image of size 256 × 256 pixels, the final S2 layer is a vector of dimension 44 × 44 × 4000. The response is obtained using

$$R(X, P) = \exp\left(\frac{\|X - P\|^2}{2\sigma^2}\right) \tag{6.5}$$

The last layer of the architecture is the global invariance layer (C2). The maximum response to each intermediate feature over all (X, Y) positions and scales is calculated. The result is a characteristics vector that will be used for classification.

For the implementation of the HMAX model, we use the tool proposed in [21]. Figure 6.9 highlights the input and the output of the HMAX in our case.

6.3.4 Classification with the Softmax Layer

The classification stage is implemented with a neural network based on the softmax function. The loss function for the softmax layer is based on the computation of the crossentropy [22]:

$$L_i = -\log\left(\frac{\mathrm{e}^{f_{y_i}}}{\sum_j \mathrm{e}^{f_j}}\right) \tag{6.6}$$

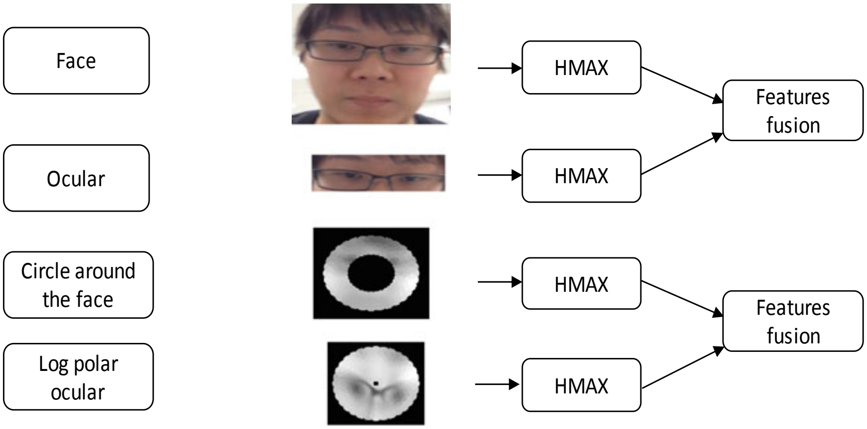


Fig. 6.9 General architecture of the proposed system based on the HMAX model

where f_j the *j*-th element of the feature vector representing subject *f*, while L_i is the full loss over the training examples.

During the training phase, feature vectors are composed of frames extracted from different video sequences from the same session. This composition of features is required to cope with the difference of illumination which is always recorded even within the same session.

6.4 Experimental Results

The UMD_AA dataset [23], a very challenging test bed for performing experiments on active authentication for mobile devices, has been used in this study. In this dataset, videos are recorded in different illumination conditions within a laboratory room. The first image subset was captured with artificial lighting. The second subset was captured without any illumination. The last subset was captured with natural sunlight. The database is composed of 45 subjects. For each subject, five videos are available for each session. One out of the five videos, containing variations in the face position and rotation, is used for testing. The remaining four videos are used for testing. The test videos are captured from mobile devices while the user is performing a specific activity, such as looking at a window popup, scrolling test, taking a picture or working on a document. In the first protocol, the training data is composed of videos pertaining to given ambient lighting while the test data belongs to the other two subsets. Therefore, there are three available scenarios for the first protocol 1 → {2, 3}, 2 → {1, 3}, 3 → {1, 2}, where 1 is the subset containing the data acquired in the indoor illumination condition, 2 is the subset containing the data acquired without illumination, and 3 is subset containing the data acquired in the sunlight illumination condition.

For the first experiment, the original HMAX model was applied to process the full faces and the ocular regions in the UMD dataset. The face and ocular regions obtained by applying the Viola–Jones and the landmark detection algorithms were used as input for HMAX network. The classification rates obtained from the ocular regions and the outer face are fused using the max rule. Table 6.2 reports the recognition rates obtained using the face, ocular regions and the fusion of the two components. S1 corresponds to videos obtained from session 1, S2 to session 2 and S3 to session 3.

Table 6.2 Recognition rates obtained from the first experiment

Session	Face and ocular regions from landmark points			Face and ocular regions from the Viola–Jones algorithm		
	Outer face	Ocular regions	Fusion	Outer face	Ocular regions	Fusion
S1	88.62	85.37	**89.43**	63.41	72.50	67.48
S2	68.24	70.27	**75.68**	55.28	47.97	48.78
S3	89.43	84.55	**91.87**	57.72	60.16	63.41

The second experiment was performed by applying a full implementation of the proposed system and performing the log-polar mapping on the ocular regions, obtained from the Viola–Jones and the landmarks detection algorithms. The obtained log-polar images were used as input for the HMAX network (Table 6.3).

The recognition rates obtained in the second experiment from the first session are graphically compared in the bar histogram depicted in Fig. 6.10.

The performance of the proposed framework is compared with the algorithms applied in [23]. The best performance obtained from Fisherfaces (FF), Sparse Representation-based classification (SRC) and Mean-Sequence SRC (MSSRC) has been used for comparison and reported in Table 6.4.

The effectiveness of the landmark detection and frame selection approach over the Viola–Jones can be noted by comparing the results reported in Table 6.2.

As it can be noted by comparing the results reported from the two experiments, and graphically shown in Fig. 6.10, the proposed system always improves the performance of the HMAX model. The improvement is due to the biologically inspired retino-cortical projection applied to the raw input data.

The results obtained by applying the UMD testing protocol 1 are aligned with the performance of the other methods (FF, SRC, MSSRC) reported in [22]. However,

Table 6.3 Recognition rates obtained from the second experiment

Session	Face and ocular regions from landmark points			Face and ocular regions from the Viola–Jones algorithm		
	Outer face	Ocular regions	Fusion	Outer face	Ocular regions	Fusion
S1	96.75	87.80	**97.56**	89.17	79.17	91.67
S2	**89.19**	71.17	87.39	58.54	64.23	64.23
S3	94.31	91.87	**95.12**	87.80	85.37	91.87

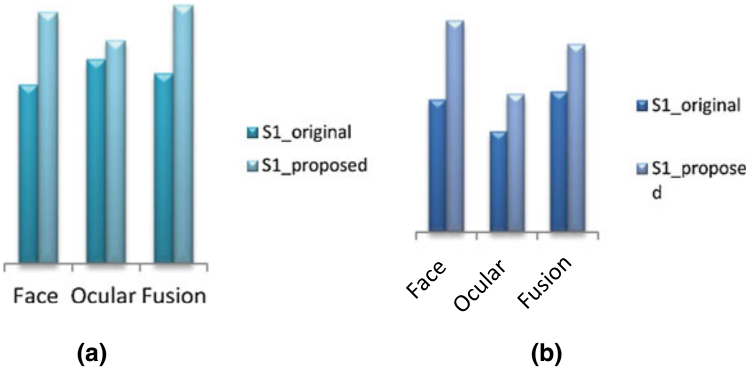


Fig. 6.10 Comparison between the recognition rate obtained by applying the original HMAX model and the proposed algorithm. **a** Recognition rates obtained by processing face and ocular regions extracted with landmark extraction and selection, **b** Recognition rates obtained by processing face and ocular regions extracted with the Viola–Jones algorithm

Table 6.4 Comparison of the results obtained by applying protocol 1 on the UMD database

Enrollment session	Test session	FF	SRC	MSSRC	Outer face	Ocular regions	Fusion
1	2	54.48	52.79	47.21	53.15	33.33	**54.95**
1	3	45.27	51.18	46.15	94.31	91.87	**95.12**
2	1	25.52	44.18	43.06	56.76	66.67	**78.38**
2	3	56.80	58.58	60.36	84.68	73.87	**84.68**
3	1	24.77	17.64	17.64	48.78	73.17	**73.98**
3	2	**56.01**	51.95	45.85	48.65	31.53	50.45

by applying a max rule fusion to the ocular and outer face scores outperformed the other methods.

6.5 Conclusion

Face recognition is now considered as a commodity available with a number of portable devices and interfaces. The high recognition performance obtained by well-engineered systems and also by the most recent deep convolutional network-based approaches is difficult to obtain. However, most of these systems are applied to challenging but not fully comparable to the viewing conditions faced in everyday life. For example, all face recognition systems operating in mobile devices require the active cooperation of the user. Moreover, the face itself has to be presented in an almost standard position, as the system is unable to operate if the face is presented rotated or upside-down. Whenever the user is not cooperative or unaware of the face being captured for instance, for continuous verification of the user's identity, the recognition performance drops dramatically.

In order to face this extremely challenging problem, a modified version of the biologically inspired HMAX model has been proposed. As the HMAX model does not take into account the retino-cortical mapping between the retinal plane and the early stages of the visual cortex in the primates, a log-polar mapping was introduced as a preprocessing step. The proposed system takes inspiration from the space-variant structure of the receptive fields in the human retina, which produces a dual image representation. Objects are represented with a very high accuracy in the fovea but within a very small field of view, while objects lying in the periphery of the retina are sampled at a very low resolution but with a wide field of view. This dual representation was reproduced by sampling image regions corresponding to the eyes with high resolution and the outer part of the face with a low-resolution log-polar sampling. From the representation produced by the HMAX model, classification is performed by including a final softmax layer.

The system performance has been assessed by processing the UMD mobile face database. From a comparative analysis, the proposed system clearly outperforms other state-of-the-art algorithms for continuous authentication.

The system proposed in this chapter was an attempt to take advantage of the biological structure of the human visual system to improve face recognition performance in very challenging environments, such as continuous authentication on mobile devices. However, further research will be required to include other features of the neural architecture of the V1 area in the brain and possibly integrate them in a unique model.

Acknowledgments This research work has been partially supported by a grant from the European Commission (H2020 MSCA RISE 690907 "IDENTITY") and by a grant of the Italian Ministry of Research (PRIN 2015).

References

1. Liao S, Jain AK, Li SZ (2013) Partial face recognition: alignment-free approach. IEEE Trans Pattern Anal Mach Intell 35(5):1193–1205
2. Weng R, Lu J, Tan YP (2016) Robust point set matching for partial face recognition. IEEE Trans Image Process 25(3):1163–1176
3. Daniilidis K (1995) Attentive visual motion processing: computations in the log-polar plan. Special issue on Theoretical Foundations of Computer Vision
4. Mutch J, Lowe D (2008) Object class recognition and localization using sparse features with limited receptive fields. Int J Comput Vis (Springer)
5. Rattani A, Derakhshani R (2018) A survey of mobile face biometrics. Comput Electr Eng 72:39–52
6. Taigman Y, Yang M, Ranzato M, Wolf L (2014) Deepface: closing the gap to human-level performance in face verification. In: IEEE international conference on computer vision and pattern recognition, pp 1701–1708
7. Sun Y, Wang X, Tang X (2014) Deep learning face representation from predicting 10,000 classes. In: IEEE international conference on computer vision and pattern recognition, pp 1891–1898
8. Schroff F, Kalenichenko D, Philbin J (2015) Facenet: a unified embedding for face recognition and clustering. In: IEEE international conference on computer vision and pattern recognition, pp 815–823
9. Liao Q, Leibo J, Poggio T (2013) Learning invariant representations and applications to face verification. In: Advances in neural information processing systems, vol 26, NIPS 2013
10. Esmaili S, Maghooli K, Nasrabadi A (2016) C3 effective features inspired from ventral and dorsal stream of visual cortex for view independent face recognition. Int J Adv Comput Sci
11. Hu X, Zhang J, Li J, Zhang B (2014) Sparsity-regularized HMAX for visual recognition. PLoS One
12. Viola P, Jones M (2001) Robust real-time object detection. In: Second international workshop on statistical and computational theories of vision, modeling, learning, computing and sampling, Vancouver, canada, 13 July 2001
13. Zhu X, Ramanan D (2012) Face detection, pose estimation, and landmark localization in the wild. In: International conference of computer vision and pattern recognition
14. Burt PG (1988) Smart sensing in machine vision. In: Machine vision: algorithms, architectures, and systems. Academic Press

15. Massone L, Sandini G, Tagliasco V (1985) Form-invariant topological mapping strategy for 2-d shape recognition. CVGIP 30(2):169–188
16. Sandini G, Tistarelli M (1991) Vision and space-variant sensing. In: Wechsler H (ed) Neural networks for perception: human and machine perception. Academic Press
17. Keith D (2012) Alliance for lighting information, foveal, para-foveal and peripheral vision, March 2012
18. Schwartz E (1984) Anatomical and physiological correlates of visual computation from striate to infero-temporal cortex. IEEE Trans Syst Man Cybern SMC-14(2):257–271
19. Grosso E, Lagorio A, Pulina L, Tistarelli M (2014) Towards practical space-variant-based face recognition and authentication. In: International workshop on biometrics and forensics, October 2014
20. http://www.vision.caltech.edu/html-files/archive.html
21. http://maxlab.neuro.georgetown.edu/hmax.html
22. CS231n: Convolutional neural networks for visual recognition, Spring 2018
23. Fathy M, Patel V, Chellappa R (2015) Face-based active authentication on mobile devices. In: IEEE international conference on acoustics, speech and signal processing (ICASSP)

Chapter 7
Selfies for Mobile Biometrics: Sample Quality in Unconstrained Environments

Chiara Lunerti, Richard Guest, Ramon Blanco-Gonzalo and Raul Sanchez-Reillo

Abstract Taking a ‘selfie’ using a mobile device has become a natural gesture in everyday life. This simple action has many similarities to face authentication on a smartphone: positioning the camera, adjusting the pose, choosing the right background and looking for the best lighting conditions. In the context of face authentication, most of the standardised processes and best practice for image quality is mainly focused on passport images and only recently has the attention of research moved to mobile devices. There is a lack of an agile methodology that adapts the characteristics of facial images taken on smartphone cameras in an unconstrained environment. The main objective of our study is to improve the performances of facial verification systems when implemented on smartphones. We asked 53 participants to take a minimum of 150 ‘selfies’ suitable for biometric verification on an Android smartphone. Images were considered from constrained and unconstrained environments, where users took images both in indoor and outdoor locations, simulating real-life scenarios. We subsequently calculated the quality metrics for each image. To understand how each quality metric affected the authentication outcome, we obtained biometric scores from the comparison of each image to a range of images. Our results describe how each quality metric is affected by the environment variations and user pose using the biometric scores obtained. Our study is a contribution to improve the performance and the adaptability of face verification systems to any environmental conditions, applications and devices.

C. Lunerti (✉) · R. Guest
University of Kent, Canterbury CT2 7NT, Kent, UK
e-mail: c.lunerti@kent.ac.uk

R. Guest
e-mail: r.m.guest@kent.ac.uk

R. Blanco-Gonzalo · R. Sanchez-Reillo
University of Carlos III of Madrid, Leganés, Spain
e-mail: rbgonzal@ing.uc3m.es

R. Sanchez-Reillo
e-mail: rsreillo@ing.uc3m.es

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_7

7.1 Introduction

Mobile devices have brought a significant change in everyday life. They are ubiquitous both for business and personal tasks including storing sensitive data and information; from saving images to a photo gallery to interacting with financial information. As such, and given the mobile nature of the devices, data have the risk of being accessed by unauthorised users. It is therefore of critical importance to secure mobile devices through appropriate and effective authorisation processes.

Personal identification numbers (PIN) and passwords are two techniques that have been traditionally used to protect access to a mobile device across a range of mobile device manufacturers and operating systems (OSs). In 2008, the Android OS also introduced a personalised graphical pattern system that allows the unlocking of the device by the connection of at least four dots on a 3 × 3 grid. However, all these security methods are vulnerable to attacks such as shoulder surfing and latent finger traces or are easy to replicate or guess [1, 2].

Biometrics has quickly become a viable alternative to traditional methods of authentication. The use of biometric verification technologies provides many advantages as the authentication is achieved using a personal aspect that users do not need to remember and that is impossible to lose. Adoption of authentication using face images as a security mode began in 2011 when Google introduced in Android 4.0 ‘Ice Cream Sandwich’ a face verification system called face unlock. In recent years, the system has updated and improved. Now called Trusted Face, starting with Android 5.0 ‘Lollipop’, it has been included as part of the smart lock system [3]. In November 2017, Apple Inc. released the iPhone X with FaceID, a verification system that works with a TrueDepth camera system. This technology comprises an infrared camera, a dot projector and a flood illuminator, with a claim to allow high face verification performances even in hostile light condition and robust against facial changes like growing hair and beard [4].

To authenticate on a mobile facial verification system, users need to take a self-portrait using the front-mounted camera of the device. Since this action corresponds to the definition of ‘taking a selfie’, it is possible to identify the relationship between the process of selfie generation and smartphone authentication. However, we can identify substantial differences between these processes depending on the use context. For instance, to ensure a successful authentication, the selfie should not be taken with other people, as this would add additional processing to the system for selecting the appropriate face to authenticate among the others. Also, the facial expression should be neutral, to avoid variability on the image.

Despite these differences, it is possible to surmise that the massive popularity of posting selfies on social media has helped with the acceptability of mobile face verification. The growth of the use of facial systems on mobile devices has not been without issues. According to a survey of 383 subjects conducted by De Luca et al. in 2015, a shift was observed as to the motivations to cause people to abandon the use of face unlock, primarily from overriding privacy concerns to social compatibility. Across the subjects, 29% declared that they stopped using face unlock for usability concerns (such as variable performance caused by environmental problems) and for the feeling of awkwardness in taking a selfie in front of other people for authentication [5].

The recent acceptance in the social context of taking selfies in public is playing an essential role in the acceptability of face verification on a smartphone, leading to the socially acceptable possibility of selfie authentication or selfie banking. In work presented by Cook [6] in 2017, the authors underline that an increasing number of users are checking their bank accounts using their mobile devices, and they are willing to use face verification as a biometric over other modalities, such as fingerprint, as they considered it more reliable and, through liveness detection, more secure.

It is, however, necessary to understand how taking authentication images in an unconstrained environment influences the quality (and consequently the performances) of a verification system. In face verification, most implementation standards and best practices are focused on the use of facial images in specific scenarios, such as electronic IDs or passports. Best practice needs to be adapted to the additional unconstrained environment parameters that the device mobility introduces. As the user moves the device in an unconstrained manner, both posture and the background may be subject to significant change. Also, the resolution of a device camera is typically lower than those used for taking passport images, so the same quality metrics may not have the same effect in this scenario. In the context of mobile devices, it is crucial to asses a realistic scenario including the variability of unconstrained environments.

Our research aims to contribute to the improvement of the performance of facial verification systems when applied in smartphones. We have analysed how image quality changes in respect to unconstrained environments and what influence this has on the biometric match scores. We also have studied how the user and the smartphone camera introduce variability in the system.

7.2 Biometric Selfies, the Challenges

The ISO/IEC 19794-5:2011 Biometric data interchange formats—Part 5: Face image data standard [7] provides a series of measures and recommendations to consider when collecting images for facial verification. The standard includes the acquisition process, where subjects should be in a frontal position, at a fixed distance from the camera. Images taken in unconstrained environments are mainly influenced by the different postures that users present towards a camera that is considerably smaller in size compared to the single-lens reflex (SLR) system generally used for capturing passport images. Mobile devices can also be moved, varying the distance between the subject and the capturing device, resulting in a variation of light and posture. Some existing studies [8, 9] have aimed to improve performance across different lighting conditions and poses of subjects, although the majority focus on video surveillance recognition or passport image application. In the first case, high-quality equipment is usually adopted, and in the second scenario, there is controlled variability in pose and lighting that limits the application in real-life scenarios.

One approach to enhance sample quality of a biometric system is to provide real-time feedback to subjects so that they can adjust the device or posture, or they can provide another sample. In work presented by Abaza et al. [10], the authors analysed common metrics used to assess the quality and presented an alternative face image quality measure to predict the matching performance, requesting another sample in the case where a donated image did not conform to quality requirements. The method presented by the authors was to filter low-quality images using a proposed face quality index, resulting in an improvement of the system performance from 60.67 to 69.00% when using a distribution-based algorithm (local binary patterns) and from 92.33 to 94.67% when using commercial software (PittPatt).

Another approach when dealing with low-quality images is presented by Kocjan and Saeed [11]. Their methodology consists of determining fiducial face points that are robust to different light and posture conditions by using Toeplitz matrices. Their algorithm achieved a 90% success rate when verifying images in unconstrained environments although this only occurred for a database with less than 30 users. Future research is focusing on maintaining the success rate while increasing the database size.

There are few studies explicitly focused on mobile devices. A study on smartphone and image quality [12] collected 101 subjects' images of which 22 samples from each person was captured from two different devices: a Samsung Galaxy S7 and an Apple iPhone 6 Plus during two sessions. The variation of the light position and pose of the user were fixed as participants were asked to take two images with a different yaw posture (head turn to the right or the left) and six more variating their posture with roll and pitch (head tilt to the right or the left and the back or the front, respectively). The quality was assessed over the collected database using different schemes, and the method proposed by the authors resulted in nearly equal or better performances to the other quality assessment methodologies.

Several databases have been released to assess face verification/identification covering a series of problems and challenges that this modality needs to overcome (for example, the 'Labeled Faces in the Wild' [13] database of unconstrained facial images, formed of 13,233 images from 5749 subjects taken in different light conditions and environments). However, there is a lack of a suitable unconstrained environment facial image database with samples taken from a smartphone. Available databases usually focus on a specific environment such as an office or a laboratory and with controlled movements and posture for the user.

The main contribution of our study is the analysis of selfie biometrics considered in real-life scenarios where the unconstrained environment introduces variations in quality, interaction and performances. This work builds on our previous study [14] where we described the quality variations in constrained and unconstrained environments considering quality metrics conformant to the standard requirements for passport images.

7.3 Data Collection

With the aim of assessing the impact that different types of environments have on selfies for mobile verification, we carried out an analysis by undertaking our data collection. We designed a collection process lasting about 30 min repeated across three time-separated sessions where participants took selfies suitable for verification on a provided mobile device (a Google Nexus 5). Full local ethics approval was granted prior to the commencement of our data collection.

During the first session, participants were informed as to the nature of the study and demographics were recorded. Information was also recorded regarding participants' previous experience with biometric systems and biometric authentication on mobile devices. Following this process, they received an explanation on smartphone enrolment. Each participant was asked to sit on a chair at a fixed distance from the camera (2 m) in a room with only artificial light and a white background. Six pictures were taken by a supervisor using a Canon EOS 30D SLR following the specification for passport images as described in the standard ISO/IEC 19794-5. Under the same conditions, they were given the smartphone and were asked to take another five images by themselves using the front-mounted camera of the Nexus 5 and this provided data to compare the *ideal conditions* of enrolment across two different cameras.

For the remainder of Session 1, and for the following two sessions, a standard procedure was followed. Participants were required to follow a map of locations where they were to capture a minimum of 5 verification images. The map differed across each capture session. Each map contained a total of 10 locations resulting in a minimum of 150 selfies for each participant. The locations varied: indoors and outdoors, crowded and less crowded, and were representative of locations where smartphones are used in everyday life (cafés, car parks, corridors of a building, etc.).

To collect all the images, we used an Android app that was developed for this study which also helped the participants to keep the count of the number of selfies taken during the session. The only instruction that participants received was to take the selfies for verification: ideally, they were advised to present a neutral expression and a frontal pose to the camera, but they were free to move as required, assessing lighting conditions and background that, in their opinion, was ideal to provide their biometrics for verification. We collected a total of 9728 images from 53 participants of which only one participant did not complete all three sessions. Gender of participants was balanced (50.5% F/49.5% M).

7.4 Data Analysis

Based on the research questions that we wished to address, we considered our analysis according to the diagram shown in Fig. 7.1. The figure shows the contributory variables that we wanted to investigate, and their relationships are indicated by the

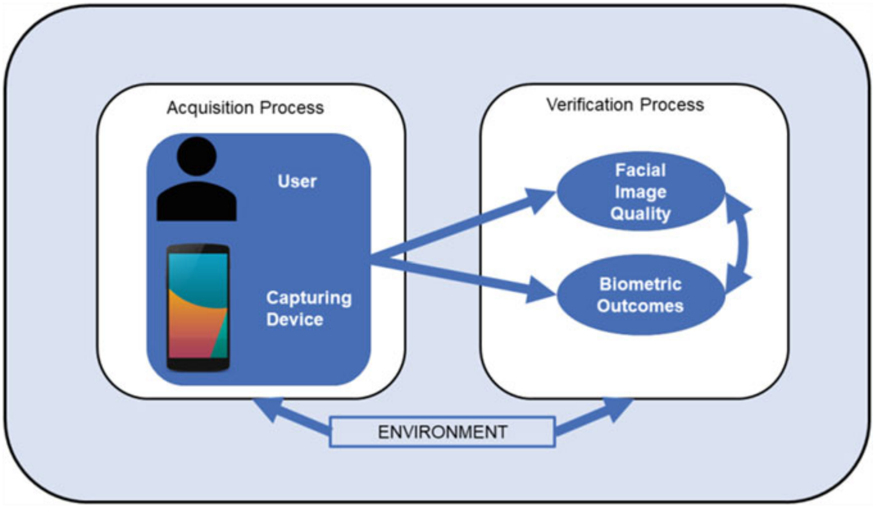


Fig. 7.1 Diagram of relationships considered in a mobile face verification system

arrows. These relationships can be explored across different types of environment. The acquisition process in mobile scenarios is not a fixed system. Both the user and the smartphone can move freely. In the verification process, Facial image quality and biometric outcome scores receive influence from the user interaction and the capturing sensor. All variables are under the influence of different environments.

7.4.1 Biometric Verification

We first used two different algorithms to assess facial detection, Viola–Jones [15] as an open-source algorithm that is commonly used for this task, and the detection system with a state-of-the-art commercial verification system [16]. The commercial biometric system (CBS) was also used to assess biometric verification performance.

We considered four enrolment scenarios. The first enrolment (E1) included five images taken using the SLR camera under static conditions as previously explained. Under the same static condition, the second type of enrolment (E2) used images taken with the smartphone camera. These first two types of enrolment enabled a comparison of different types of cameras under the same ideal enrolment conditions.

The other two types of enrolment replicate real-life situations where the user is using the face authentication for the first time and need to enrol on the smartphone. We selected five random images taken indoors for the third enrolment (E3) and five random images from the images taken outdoors (E4). We decided to exclude a random combination between images taken indoors and outdoors because we assumed that it would be unlikely that someone will change his or her location from indoors to outdoors (or vice versa) in this situation.

Once all the images had been selected for the enrolment, we then considered all remaining images from that participant for verification. We used the CBS to perform the biometric verification, recording a failure to detect when the CBS could not recognise a face within an image. We calculated a biometric score (BS) as the mean of the comparisons of one verification image against all five enrolment images and a biometric outcome (BO) as either ‘succeeded’ or ‘failed’ depending on the majority between the five comparisons.

7.4.2 The User

The user can introduce two types of influencing factors. Some characteristics are intrinsic to the participant (such as demographic characteristics) and others that can be temporary (such as glasses, type of clothing and facial expression). From the demographics, we considered age, gender and previous experience (both with biometrics in general and in biometrics used on a mobile device) that the users declared before taking part in the experiment. We wanted to verify that there were not any differences in terms of quality and performance assessment within any demographic groups.

We used the CBS to estimate the facial expression that the user made during the image acquisition concerning the level of anger, disgust, fear, happiness, neutral, sadness and surprise. Each expression is recorded as a percentage of confidence that the user exhibits a particular expression in a captured image.

7.4.3 The Capture Device

The capture devices used during the data collection were a Canon EOS 30D SLR and a Nexus 5 smartphone camera. We provided the same model of mobile device to all the participants, to ensure that there were no differences regarding camera resolutions between the images. This decision had been made to obtain results that are device-independent and that the observations made in this study are generally valid in any case of scenarios.

We hypothesised that the images taken with the SLR would be higher-quality images and that it would be easier to use for verification over a lower-quality image taken from a smartphone camera. The camera specifics for both types of devices are summarised in Table 7.1.

The exchangeable image file format (Exif) file, providing information related to the image format, was examined from each image to establish the variation capture equipment. Recent phones allow the owner to access, personalise and modify specific

Table 7.1 Camera specifics for the SLR Canon EOS 30D and the Google Nexus 5 cameras used during the data collection [17, 18]

Camera specifics	Canon EOS 30D	Google Nexus 5
Type	Digital AF/AE SLR	Selfie camera
Pixels	8.5 MP	1.3 MP
Focal length (35 mm)	35 mm	33 mm
Sensor pixel size	22.5 × 15.0 mm	1.95 μm
Autofocus features	Autofocus 9 point	Fixed focus

characteristics of the frontal camera but with the Nexus 5 that was not possible, and the focus was set to automatic.

The main camera settings that give control over quality are the aperture, ISO and shutter speed [19]. Aperture is the size of the hole within the lens that controls the lights that enters the camera body and consequentially the focus of the subject. In our experiment, it had a fixed value of 2.9 throughout all the images taken with both the smartphone camera and the SLR. Shutter speed is the length of time the camera shutter opens when taking the image. The SLR camera was fixed in position with a tripod, and the shutter speed was set at 1/60 recording images of ideally not moving subjects. When taking selfies with the smartphone, not only the subjects are moving but also the camera can take a different position, depending on how the user is holding the device. It becomes hard to differentiate these types of movements, and for this reason, the settings that we decided to consider in our analysis is the variation in ISO that measures the sensitivity of the camera sensor. The SLR had a fixed value set to 400, while the smartphone camera ISO variates between 100 and 2000.

7.4.4 *Environment*

We considered two types of environmental conditions. The experiment room, where there was only a fixed artificial light and participants were sitting on a chair with a white background, presented an indoor environment with ideal conditions. Images taken in this scenario were collected using both the SLR and the smartphone camera (SmrC).

All the selfies taken with the smartphone outside the experiment room have been collected in unconstrained environmental conditions. We analysed separately the images taken in the unconstrained environment when outdoors and when indoors.

7.4.5 *Facial Image Quality Metrics*

To assess the facial quality of the selfies acquired during the data collection, we followed the recommendations of ISO/IEC TR 29794-5 Technical Report (TR) [20].

Out of the several facial image quality (FIQ) metrics considered in the TR, we selected five metrics as the ones that are commonly used in the state-of-the-art to describe quality features. Image brightness refers to the overall lightness or darkness of the image. The image contrast helps to understand the difference in brightness between the user and the background of the image. The global contrast factor (GCF) determines the richness of contrast in details perceived in an image. The higher the GCF, the more detailed the image. Image blur quantifies the sharpness of an image. Finally, the exposure quantifies the distribution of the light in an image.

Below, there is a description on how to calculate each FIQ metric:

Image Brightness (*B*)

Image brightness is a measure of pixels intensities of an image. As defined in the TR, the image brightness can be represented by the mean of the intensity values h_i, where $i \in \{0, \ldots, N\}$.

The mean of the histogram $\bar{h}$ can be represented by the formula:

$$\bar{h} = \frac{1}{N+1} \sum_{i=0}^{N} h_i$$

where h is the intensity value of each pixel, and N is the maximum possible intensity value.

Image Contrast (C)

Image contrast is the difference in luminance of the object in the image. There are different ways to define image contrast—we chose to calculate it from the histogram of the whole image using the following formula:

$$C = \sqrt{\frac{\sum_{x=1}^{N} \sum_{y=1}^{N} (I(x, y) - \mu)^2}{MN}}$$

where $I(x, y)$ is the image face of size $M \times N$, and μ represents the mean intensity value of the image.

Global Contrast Factor (GCF)

The global contrast factor (GCF) is described in the TR as the sum of the average local contrasts for different resolutions multiplied by a weighting factor. We calculated the GCF following the methodology presented by Matkovic et al. [21]. The local contrast is calculated at the finest resolution that is the original image as the average difference between neighbouring pixels. Then the local contrast is calculated for various resolutions that are obtained combining four original pixels into one super pixels, reducing the image width and height to half of the original ones. This process has been done for a number of R iterations. The global contrast is then calculated as a weighted average of local contrasts:

$$\mathrm{GCF} = \sum_{k=1}^{R} w_k C_k$$

where C_k is the local contrast for R a number of resolutions considered, and w_k is the weighting factor. The authors defined the optimum approximation for the weighting factor over R resolution levels as:

$$w_k = \left(-0.406385\frac{k}{R} + 0.334573\right)\frac{k}{R} + 0.0877526$$

where w_k ranges from 1 to the number of resolutions (R) of the image considered.

Image Blur (Blur)

To calculate the blur effect, we studied the work presented by Crete et al. [22]. Their methodology allows the determination of a no-reference perceptual blurriness of an image by selecting the maximum blur among the vertical direction $\mathrm{blur}_{\mathrm{ver}}$, and the one among the horizontal one $\mathrm{blur}_{\mathrm{hor}}$.

$$\mathrm{Blur} = \mathrm{Max}(\mathrm{blur}_{\mathrm{ver}}, \mathrm{blur}_{\mathrm{hor}})$$

The metric range is between 0 and 1, where 0 is the best and 1 is the worst quality.

Exposure (E)

Exposure can be characterised by the degree of distribution of the image pixels over the greyscale or over the range of values in each colour channel. As defined in the TR, exposure can be calculated as a statistical measure of the pixel intensity distribution, such as entropy [23].

$$E = -\sum_{i=1}^{N} p_i \log_2 p_i$$

where p_i is the histogram of the intensity level for the N possible intensity levels.

7.5 Results

As a pre-processing stage, we removed the images that were taken by mistake (for example, that did not include a facial image, or contained other people), obtaining a final database of 9420 selfie images. In this paragraph, we illustrate the results obtained according to the different elements considered for image quality, biometric outcomes and user expressions.

7.5.1 Image Quality

Our initial investigation was to understand the variations regarding the quality of facial images. We wanted to assess how each metric varies depending on the many factors that affect the system, including different types of environments.

From Table 7.2, we can observe that the original means have around the same values as the median, so we can assume that extreme scores do not influence the mean. A further analysis assessing the 5% trimmed means confirmed that there were no substantial outliers in the distribution that affect the mean values. From the skewness and kurtosis analysis, we can ascertain that all the variables are normally distributed, as their values are between -1.96 and 1.96, except esxposure (E).

We studied the quality metrics under different conditions. Since each FIQ metric has a different range of values, we analysed them separately to understand their relationship with the user and the type of environmental conditions. In Fig. 7.2, we can see the variations of image brightness (B) across the 53 participants. This feature could be used to distinguish the images that have been taken in ideal conditions from the ones taken in the unconstrained environment. The threshold that is presented in the graph, as well as in the following figures that describe each quality metrics, represents an example of an empirically selected threshold (120) that can be used to distinguish between images taken in a constrained or unconstrained environment. A further study needs to be carried out to determine the optimal thresholds that could be generally valid for any type of camera sensors.

The images that have been taken with the SLR in static condition have quality values different from those taken with a smartphone camera in unconstrained environments, indicated separately for indoors and outdoors location and the distinction between static conditions when using the smartphone is less evident. For SLR images, B ranges between 120 and 160 while for images taken indoors and outdoors in the unconstrained environments the range is from 90 to 120. When investigating brightness considering additional influencing factors, we observed that the values appear to be stable across all the three sessions and there are no significant differences between gender and age. Similarly, people that had previous experience with (mobile) biometrics did not result in different images concerning brightness compared to those who had not experienced biometric systems.

From Fig. 7.3, we can see the variation in image contrast (C) across all the participants. In this case, SLR images taken in ideal conditions vary across the users with values from around 11–13, while in unconstrained scenarios, the images presented values with variation from 9.5 to 11. C provides a clearer division compared to B between ideal conditions and unconstrained environment. No significant differences were identified across demographics.

Contrary to the previous two FIQ metrics, GCF calculated on SLR images, as shown in Fig. 7.4, appears centred between a small range (from 1 to 3) compared to the values of all the images taken by the smartphone.

All the images captured using the smartphone range from 3 to 6.5, including those under ideal conditions, making impossible to distinguish them from the uncon-

Table 7.2 descriptive statistics of each FIQ metrics for the whole database (9420 selfies)

FIQ metrics	Min	Max	Mean	Median	Std. dev.	Skewness		Kurtosis	
						Stat.	Std. err	Stat.	Std. err.
B	21.19	210.28	108.81	107.98	15.56	0.052	0.025	1.822	0.05
C	6.52	13.79	10.39	10.29	0.822	0.552	0.025	0.624	0.05
GCF	1.05	9.60	5.19	5.19	1.32	−0.256	0.025	0.361	0.05
Blur	0.18	0.49	0.30	0.29	0.042	0.496	0.025	−0.071	0.05
E	5.01	7.98	7.53	7.6	0.29	−1.65	0.025	4.21	0.05

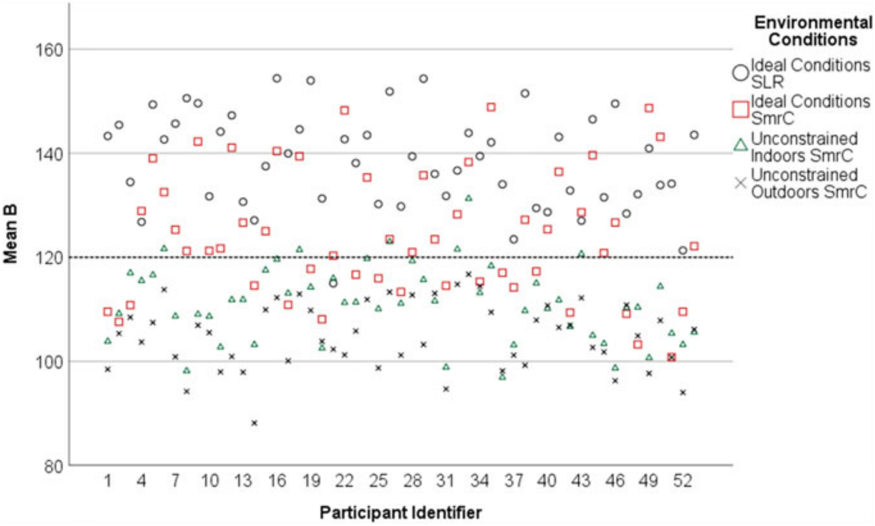


Fig. 7.2 Mean values of Image Brightness across 53 participants

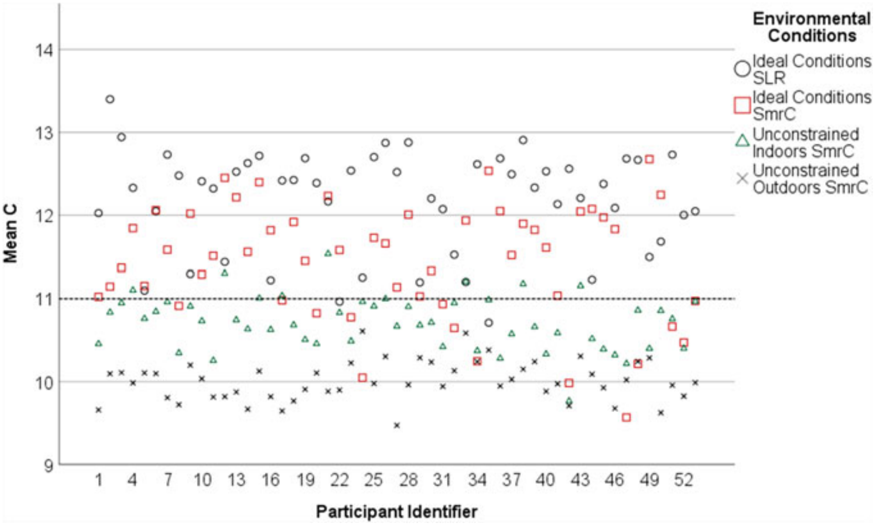


Fig. 7.3 Mean values of image contrast across 53 participants

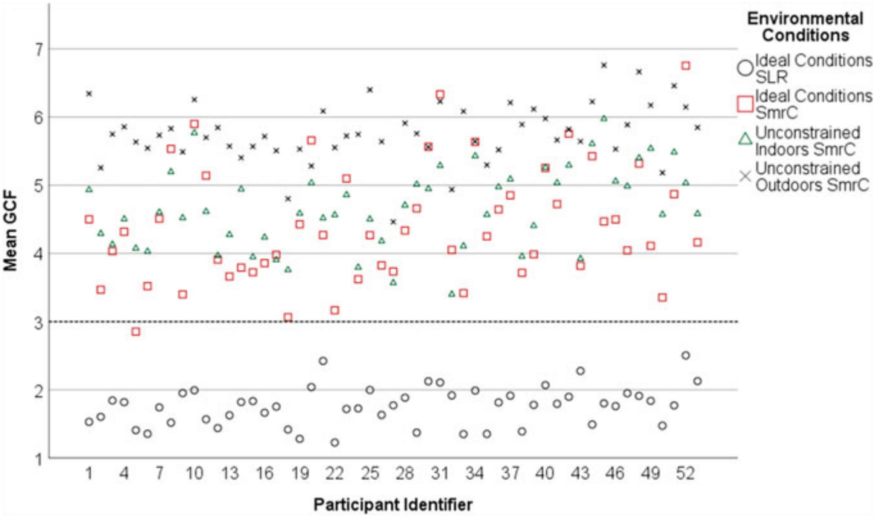


Fig. 7.4 Mean values of GCF across 53 participants

strained environment. GCF resulted in the only quality metric considered that is influenced by the demographic. There is a negative correlation with age ($r = -0.123$, $n = 9728$, $p < 0.001$). It looks like younger participants tented to take images with higher GCF, hence more high defined images. This could be of interest for future analysis.

Like GCF, image blur (Fig. 7.5) also presented a distinct range of values for images taken with the SLR compared to when using the smartphone camera under the same ideal conditions. Across the collected facial images, there were not many cases of an extreme blur—all the participant reported blurriness less than 0.36. Ideal conditions with the SLR can be detected from having a range of values less than 0.26, while all the images taken with the mobile device range between 0.26 to 0.36. Even though it could be unclear to form a distinction between images taken in ideal conditions with a smartphone and those taken in the unconstrained environments, we can still notice a distinction between images taken when indoors (from 0.31 to 0.36) and outdoors (0.26–0.31). There are no differences regarding sessions, demographics and previous experience.

Exposure values (Fig. 7.6) for SLR images are between the ranges of 6.65–7.35, whereas we can put a threshold to differentiate them from smartphone images taken indoors and outdoors that range from 7.35 to 7.80, and we cannot make a distinction with the images taken in ideal conditions with the smartphone. There are no significant differences between sessions, gender and age.

We also inspected the variation of ISO when the images were taken in different environmental conditions in an attempt to analyse the correlation between the camera specifics and the levels of FIQ metrics. ISO distribution does not appear normally distributed, but from the analysis of the scatter plots, we observed a linear correlation

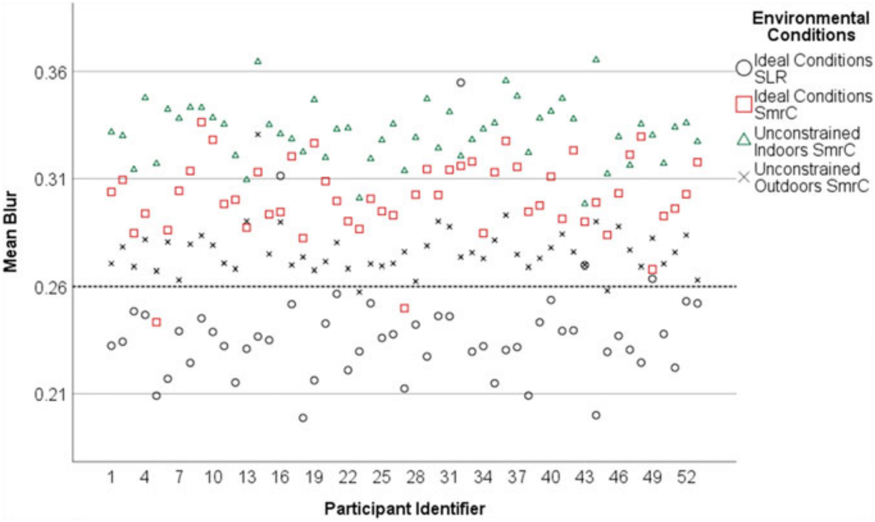


Fig. 7.5 Mean values of Image blur across 53 participants

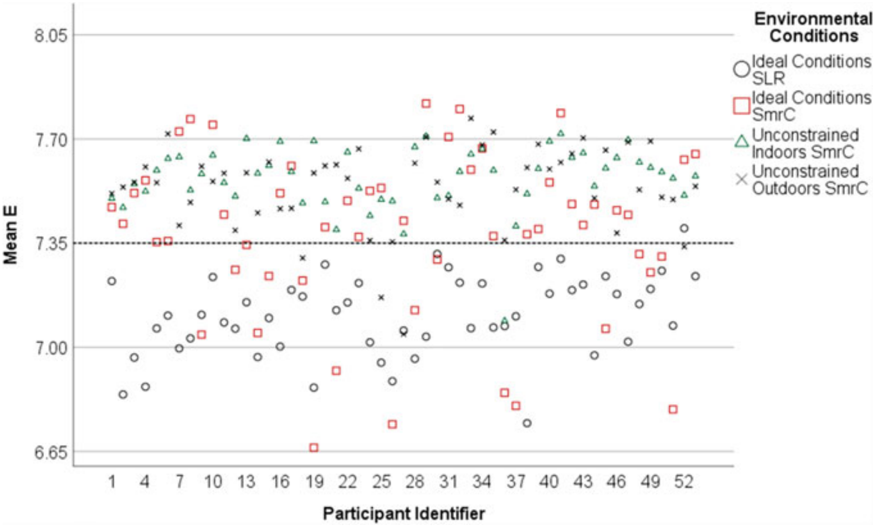


Fig. 7.6 Mean values of exposure across 53 participants

that we investigated through a nonparametric Spearman correlation. There were significant results for each of the FIQ metrics, but there was a particularly strong positive correlation for blur ($r = 0.528$, $n = 9420$, $p < 0.001$) and C ($r = 0.451$, $n = 9420$, $p < 0.001$). ISO values have a negative correlation with GCF for $r = -0.438$, $n = 9420$, $p < 0.001$. The correlation for B and E is less strong, with correspondently positive values for $r = 2.28$ and negative for $r = -0.072$ ($n = 9420$, $p < 0.001$).

Acknowledging the correlation between each quality metric and ISO specification, we can determine the required FIQ levels that we want to achieve and fix the ISO value on the capturing sensor. Alternatively, it may be possible to predict outcome in quality from the ISO value and be able to provide feedback in real time or request a new image from the user to ensure that the selfie will appear with the required quality for verification.

7.5.2 Biometric Results

To perform biometric verification, we first detect the facial area of each image in our data set. A facial area was detected within all the images taken in ideal conditions when using the SLR. Table 7.3 shows the failure to detect (FTD) using the Viola–Jones algorithm and the CBS. Overall, the number of faces detected across the entire database is above 90%. In a controlled environment, CBS was not able to detect three faces, using Viola–Jones, only one facial image was not detected. A higher percentage of FTD is recorded when images were taken outdoors (7.5% for CBS and 5.7% for Viola–Jones).

We analysed the outcomes of the biometric system depending on the type of environment. We aimed to understand how different type of environmental conditions influence the biometric outcome and if there is a relationship between quality and biometric scores. A relationship can be used to regulate a biometric threshold to adapt

Table 7.3 Frequency and percentage of FTD recorded by the two algorithms

Environmental conditions			Viola-Jones		CBS	
			Frequency	Per cent	Frequency	Per cent
Ideal conditions	Valid	FTD	1	0.4	3	1.1
		Detected	264	99.6	262	98.9
		Total	265	100.0	265	100.0
Unconstrained indoors	Valid	FTD	135	3.9	194	5.5
		Detected	3364	96.1	3305	94.5
		Total	3499	100.0	3499	100.0
Unconstrained outdoors	Valid	FTD	306	5.7	400	7.5
		Detected	5032	94.3	4938	92.5
		Total	5338	100.0	5338	100.0

Table 7.4 Percentages of succeeded and failed verification across different environmental conditions when using a smartphone

Environmental conditions	Verification Dataset	Outcome	E1	E2	E3	E4
Ideal conditions	$N = 210$	Succeeded	96.7	100	99.5	99
		Failed	3.3	0	0.5	1
Unconstrained Indoors	$N = 3040$	Succeeded	91.8	97.4	98.9	98.1
		Failed	8.2	2.6	1.1	1.9
Unconstrained Outdoors	$N = 4683$	Succeeded	88.7	96.1	97.7	99.2
		Failed	11.3	3.9	2.3	0.8

it to the different conditions and to ensure high performances in any unconstrained environments.

Table 7.4 shows the different percentages of verification success and failure for the different environments.

A higher percentage of users that have been mistakenly rejected by the system is recorded when the enrolment has been performed using the SLR images in ideal conditions (E1), particularly when the verification takes place in an unconstrained environment, where returned results of 8.2% indoors and 11.3% outdoors. Despite having a better resolution, verification comparisons between images taken from an SLR and a smartphone yield poorer results, as already observed in our previous study [14]. This outcome could result from the application of the chosen matching algorithm to two different types of camera sensors, and it highlight the importance of using an accurate cross-sensor matching in the particular scenario between static SLR images and mobile camera images. Future research should focus on addressing this issue analysing images collected using different camera sensors to study the effects that this can have on biometric performances.

Enrolment performed with a smartphone in ideal conditions (E2) obtained the perfect acceptance rate for images taken under the same conditions, as expected, but it also recorded a favourable success rate for both the type of unconstrained environments, with 97.4% for verification performed when indoors and 96.1% when outdoors.

When the enrolment has occurred within an unconstrained environment (E3 and E4), it can be seen that a system is more resilient to the different types of verification environments, meaning that it would be better to enrol ideally under conditions that are adverse in terms of light and background so that we can ensure higher performances across a broad range of environments.

To perform a correlation between biometric scores and quality metrics, we need to check whether the scores are also normally distributed. Table 7.5 shows the descriptive statistics for the biometric scores recorded during the verification of images against the four types of enrolments. Checking the skewness and kurtosis values, we can say that not all the biometric scores form a normal distribution with only a few exceptions. In the table are also reported the minimum and maximum bio-

Table 7.5 Descriptive statistics for the biometric scores recorded in different environments

Environmental conditions		Min	Max	Mean	Std. dev.	Skewness		Kurtosis	
						Stat.	Std. err.	Stat.	Std. err.
Ideal conditions	E1	14.2	219.2	128.29	44.94	−0.188	0.168	−0.614	0.334
	E2	593.8	1193.2	916.1	147.83	0.073	0.168	−1.106	0.334
	E3	47.4	281.8	95.88	31.46	2.544	0.168	11.936	0.334
	E4	43.6	175.2	92.81	29.54	0.798	0.168	−0.003	0.334
Unconstrained indoors	E1	8.4	205.8	80.3	26.51	0.644	0.044	1.023	0.089
	E2	26.0	738.0	95.40	41.10	3.721	0.044	33.865	0.089
	E3	23.8	420.6	115.09	42.29	1.859	0.044	5.714	0.089
	E4	30.2	267.0	100.53	31.52	0.890	0.044	1.523	0.089
Unconstrained outdoors	E1	4.2	239.2	76.86	27.73	1.017	0.036	1.870	0.072
	E2	4.4	262.4	92.27	37.02	1.246	0.036	1.857	0.072
	E3	7.4	198.0	99.26	30.57	0.447	0.036	−0.232	0.072
	E4	7.6	687.6	140.63	64.75	1.917	0.036	7.675	0.072

Table 7.6 Correlation between biometric scores and FIQ metrics for $n = 7923$

		BS_E1	BS_E2	BS_E3	BS_E4
B	Spearman's rho	0.028*	0.076**	0.041**	−0.130**
	Sig. (2-tailed)	0.014	0.000	0.000	0.000
C	Spearman's rho	0.053**	0.057**	0.047**	−0.222**
	Sig. (2-tailed)	0.000	0.000	0.000	0.000
GCF	Spearman's rho	−0.096**	−0.095**	−0.117**	0.202**
	Sig. (2-tailed)	0.000	0.000	0.000	0.000
Blur	Spearman's rho	0.049**	0.042**	0.105**	−0.288**
	Sig. (2-tailed)	0.000	0.000	0.000	0.000
E	Spearman's rho	−0.059**	−0.064**	−0.001	−0.027*
	Sig. (2-tailed)	0.000	0.000	0.896	0.016

*Correlation is significant at the 0.05 level (two tailed)
**Correlation is significant at the 0.01 level (two tailed)

metric scores recorded in the different environments (and their means and standard deviations).

We performed a nonparametric (Spearman) correlation shown in Table 7.6. The correlation has been performed for all the verification images ($n = 7923$) taken with the smartphone in both constrained and unconstrained environment. We investigated the correlation between the quality metrics recorded for those images and their biometric scores recorded when comparing them against the four types of enrolment.

From Table 7.6, we can observe some significant correlations, but not particularly strong overall (all values of the correlation coefficient, r, are smaller than 0.29). Image blur has a strong negative correlation with the fourth type of enrolment E4 ($r = -0.288$, $n = 7923$, $p < 0.001$). In a scenario where the enrolment is performed in an unconstrained outdoor environment, the verification images appear to be more sensitive to the blurriness of the image. The correlation indicates that a reduction of blurriness of the image corresponds to a higher biometric score during the verification. Exposure presented a weak correlation that is negative for all the type of enrolments. The other quality metrics tend to have overall a positive correlation with the first three types of enrolment (captured indoors), and a negative correlation for the fourth type of enrolment (captured outdoors).

GCF has the opposite behaviour, having negative correlations with the first three types of enrolment, and a positive correlation with the E4. This can mean that despite having higher values of GCF, hence an image richer in details, in the first three types of enrolment the performances are lower. An explanation for this could be the influence that the GCF receives from local contrast in different areas of the image. For instance, a facial image can have a lower contrast in one side of the image compared to the other one, and this cannot be recorded using the image contrast. This difference in contrast on the same image can influence the performances in the first three types of enrolment as it has been recorded to occur more frequently when the images were taken in indoor locations.

7.5.3 *User's Facial Expressions*

For most of the images taken with the SLR and the smartphone camera where it has been possible to detect a face ($n = 7888$), the CBS provided a level of confidence that the user was displaying a series of facial expression. In our study, we wanted to inspect if there is a correlation between the user's facial expressions and the quality level recorded, as well as the outcome from the biometric system, considering the variation that the different type of environmental conditions adds. In Fig. 7.7, we can see the mean of a facial expression's confidence for each environmental condition, indicating the frequency with which each specific expression occurred in different scenarios.

Users were only instructed to take selfies during the data collection that could be used for biometric authentication. The ideal posture would be frontal and with a neutral expression. So as expected, the facial expression that occurs the most is the neutral expression with a mean value above 40% across all scenarios. For images taken with the SLR under ideal conditions, a neutral expression has a confidence level of more than 60%. Another expression with a mean value of more than the 40% is '*surprise*' which notably occurred when using the smartphone camera. It was reported by the participants that in situations of inclement weather when outdoors, particularly with rain and strong wind, it had been harder for them take the selfies

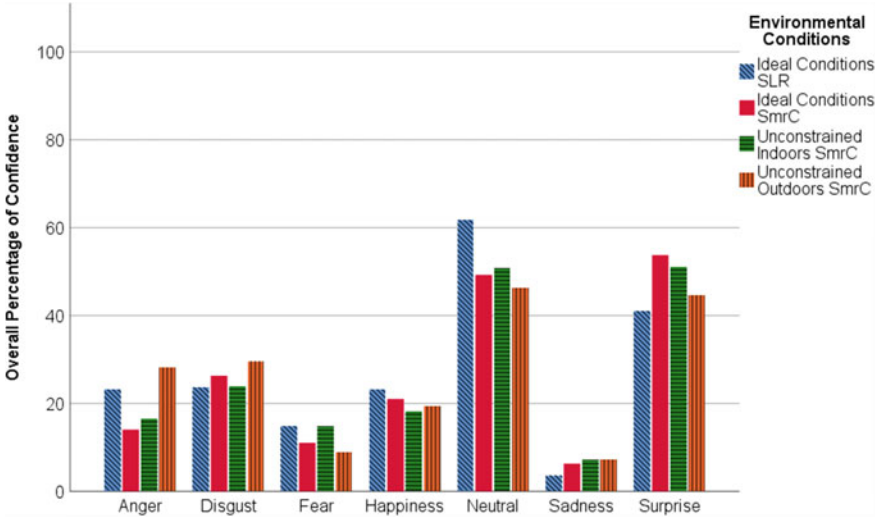


Fig. 7.7 Mean of confidence values for facial expressions

for face authentication that conformed to the requirements asked from them and this may explain why the level of disgust and anger is higher for images taken in unconstrained outdoor environment.

Facial expressions do not conform to the normality assumption for a parametric correlation, so a Spearman correlation has been used to assess the relation that different facial expressions have on both quality and biometric performances. We did not find any particularly strong correlations between quality metrics and facial expressions (the correlation coefficient was smaller than 0.18), but we did however observe a correlation with the biometric outcomes. We considered the correlation with all the verification images where it could be possible to estimate facial expressions ($n = 7678$) and their biometric scores for each of the enrolment type. We noticed a strong positive correlation for neutral expression in each enrolment scenario: under ideal conditions for images taken with the SLR ($r = 0.324$, $n = 7678$, $p < 0.001$) and the smartphone ($r = 0.318$, $n = 7678$, $p < 0.001$) and for enrolment that was performed in unconstrained environments indoors ($r = 0.382$, $n = 7678$, $p < 0.001$) and outdoors ($r = 0.295$, $n = 7678$, $p < 0.001$). Among the other facial expressions estimated, we also observed that an expression of disgust has a strong negative correlation with ideal conditions of enrolment performed with SLR ($r = -0.314$, $n = 7678$, $p < 0.001$) and the smartphone camera ($r = -0.211$, $n = 7678$, $p < 0.001$). The correlation was also negative for confidence estimation of disgust presented in the images that recorded biometric scores when compared with unconstrained enrolment scenarios for smartphone images taken indoors ($r = -0.232$, $n = 7678$, $p < 0.001$) and outdoors ($r = -0.141$, $n = 7678$, $p < 0.001$).

7.6 Conclusions and Future Work

Our study aims to contribute to improve the adaptability and the performance of mobile facial verification systems by analysing how an unconstrained environment affects quality and biometric verification score. Our experimental results describe the variations of FIQ metrics and biometric outcomes recorded under different conditions and provide recommendations for the application of selfies biometrics in real-life scenarios.

From the analysis of five different image quality metrics selected from the ISO/IEC Technical Report for image quality applied for face verification, we found that image brightness and contrast could be employed to select whether an image has been taken in a constrained or unconstrained environment. Global contrast factor, image blur and exposure were not showing different values for ideal and unconstrained conditions as clearly as the other metrics. However, by observing the local contrast and the level of blurriness, it could be possible to observe a difference between images taken in the unconstrained environments when indoors from when outdoors. These interesting results are encouraging and lead to further investigation to assess if there are significant differences between the FIQ metrics values across each type of environments. To have an overall and realistic perspective, future research will focus on analysing results collecting images using a range of different model of devices to ensure that these overall observations can be applied in context with any possible camera model. A further experiment will also be performed to explore deblurring techniques that can improve the biometric performances on those images that presented lower-quality characteristics.

Our results also suggest that it is possible to consider camera specification to regulate the quality requirement for facial images when taken on a smartphone. From our study, our recommendations will be considering fixing a value for the ISO that can result in the FIQ desired, or to inspect the variation of ISO values to regulate the thresholds of acceptance of images before verification and request an additional presentation in case of non-compliance of the requirements for quality.

Studying the biometric scores, we can confirm that enrolment under unconstrained conditions ensures the system to be more robust against the variations of the environment regarding verification performances. We reported a linear correlation between quality and biometric scores, although not particularly strong.

The type of the environment is one of the factors that influence users' facial expressions. While there was not a significantly strong correlation between different facial expressions and the quality metrics, we reported positive and negative correlations depending on the type of expressions that affect the biometric outcomes. Future research can use this information to adapt biometric systems depending on the estimation of facial expressions detected in both the enrolment and verification scenarios considering the environment in which the interaction is taking place. The biometric system could send adapted feedbacks when the estimation of the location is possible to remind the user to maintain a neutral expression during the verification process.

References

1. Bonneau J (2012) The science of guessing: analyzing an anonymized corpus of 70 million passwords. In: 2012 IEEE symposium on security and privacy (SP), pp 538–552
2. Uellenbeck S, Dürmuth M, Wolf C, Holz T (2013) Quantifying the security of graphical passwords: the case of android unlock patterns. In: Proceedings of the 2013 ACM SIGSAC conference on computer & communications security, pp 161–172
3. Android (2018) Smart lock [Online]. Available: https://get.google.com/smartlock/
4. Apple Inc. (2018) About face ID advanced technology [Online]. Available: https://support.apple.com/en-gb/HT208108
5. De Luca, A, Hang A, Von Zezschwitz E, Hussmann H (2015) I feel like I'm taking selfies all day!: towards understanding biometric authentication on smartphones. In: Proceedings of the 33rd annual ACM conference on human factors in computing systems, pp 1411–1414
6. Cook S (2017) Selfie banking: is it a reality? Biomet Technol Today 3:9–11
7. ISO/IEC 19794-5:2011—Information technology—biometric data interchange formats—part 5: face image data [Online]. Available: http://www.iso.org/iso/catalogue_detail.htm?csnumber=50867
8. Sang J, Lei Z, Li SZ (2009) Face image quality evaluation for ISO/IEC standards 19794-5 and 29794-5. In: International conference on biometrics, no. Springer, Berlin, pp 229–238

9. Haghighat M, Abdel-Mottaleb M, Alhalabi W (2016) Fully automatic face normalization and single sample face recognition in unconstrained environments. Expert Syst Appl 47:23–34
10. Abaza A, Harrison MA, Bourlai T, Ross A (2014) Design and evaluation of photometric image quality measures for effective face recognition. IET Biomet 3(4):314–324
11. Kocjan P, Saeed K (2012) Face recognition in unconstrained environment. In: Biometrics and Kansei engineering, no. Springer, New York, pp 21–42
12. Wasnik P, Raja KB, Ramachandra R, Busch C (2017) Assessing face image quality for smartphone based face recognition system. In: 2017 5th international workshop on biometrics and forensics (IWBF), no. IEEE, pp 1–6
13. Huang GB, Mattar M, Berg T, Learned-Miller E (2008) Labeled faces in the wild: a database for studying face recognition in unconstrained environments. In: Workshop on faces in 'Real-Life' images: detection, alignment, and recognition
14. Lunerti C, Guest RM, Blanco-Gonzalo R, Sanchez-Reillo R, Baker J (2017) Environmental effects on face recognition in smartphones. In: 2017 international Carnahan conference on security technology (ICCST), no. IEEE, pp 1–6
15. Viola P, Jones MJ (2004) Robust real-time face detection. Int J Comput 57:137–154
16. NEUROtechnology VeriLook 10.0 Standard SDK [Online]. Available: http://www.neurotechnology.com/verilook.html
17. Canon, EOS 30D specifications [Online]. Available: http://web.canon.jp/imaging/eos30d/spec/index.html
18. Ubergizmo, Google Nexus 5 specifications [Online]. Available: https://www.ubergizmo.com/products/lang/en_us/devices/nexus-5/
19. Hawkins M (2017) The exposure triangle: aperture, shutter speed and ISO explained, 12 July 2017 [Online]. Available: https://www.techradar.com/uk/how-to/photography-video-capture/cameras/the-exposure-triangle-aperture-shutter-speed-and-iso-explained-1320830
20. ISO/IEC TR 29794-5:2010—Information technology—biometric sample quality—part 5: face image data [Online]. Available: https://www.iso.org/standard/50912.html
21. Matkovic K, Neumann L, Neumann A, Psik T, Purgathofer W (2005) Global contrast factor-a new approach to image contrast. Comput Aesthetic pp 159–168
22. Crete F, Dolmiere T, Ladret P, Nicolas M (2007) The blur effect: perception and estimation with a new no-reference perceptual blur metric. In: Human vision and electronic imaging XII, vol 6492, no. International Society for Optics and Photonics, p 64920I
23. Gonzalez RC, Eddins SL, Woods RE (2004) Representation and description. In: Digital image processing using MATLAB. Prentice Hall, Upper Saddle River, pp 465–466 (Chapter 11)

Part II
Selfie and Liveness Detection

Chapter 8
Presentation Attack Detection for Face in Mobile Phones

Yaojie Liu, Joel Stehouwer, Amin Jourabloo, Yousef Atoum and Xiaoming Liu

Abstract Face is the most accessible biometric modality which can be used for identity verification in mobile phone applications, and it is vulnerable to many different presentation attacks, such as using a printed face/digital screen face to access the mobile phone. Presentation attack detection is a very critical step before feeding the face image to face recognition systems. In this chapter, we introduce a novel two-stream CNN-based approach for the presentation attack detection, by extracting the patch-based features and holistic depth maps from the face images. We also introduce a two-stream CNN v2 with model optimization, compression and a strategy of continuous updating. The CNN v2 shows great performances of both generalization and efficiency. Extensive experiments are conducted on the challenging databases (CASIA-FASD, MSU-USSA, replay attack, OULU-NPU, and SiW), with comparison to the state of the art.

8.1 Introduction

Biometrics authentication systems aim to utilize physiological characteristics, such as fingerprint, face, and iris, or behavioral characteristics, such as typing rhythm and gait, to uniquely identify an individual. As biometric systems are widely used in real-world applications including unlocking cell phone and granting mobile

Y. Liu · J. Stehouwer · A. Jourabloo · Y. Atoum · X. Liu (✉)
Department of Computer Science and Engineering,
Michigan State University, East Lansing, MI 48824, USA
e-mail: liuxm@msu.edu

Y. Liu
e-mail: liuyaoj1@msu.edu

J. Stehouwer
e-mail: stehouw7@msu.edu

A. Jourabloo
e-mail: jourablo@msu.edu

Y. Atoum
e-mail: atoumyou@msu.edu

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision
and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_8

transaction, biometric spoofs, or presentation attacks (PA) are becoming a large threat, where a spoof biometric sample is presented to the biometric system and attempts to be authenticated. Face, as the most accessible biometric modality, has many different types of PAs including print attack, replay attack, 3D masks, etc. As a result, conventional face recognition systems can be very vulnerable to such PAs and are exposed to risk and loss beyond measure.

In order to develop a face recognition system that is invulnerable to various types of PAs, there is an increasing demand in designing a robust presentation attack detection (PAD or face anti-spoofing) system to classify a face sample as live/spoof *before* recognizing its identity. Previous approaches to PAD can be categorized into three groups. The first is the texture-based methods, which discover discriminative texture characteristics unique to various attack mediums. Due to a lack of an explicit correlation between pixel intensities and different types of attacks, extracting robust texture features is challenging. The second is the motion-based methods that aim at classifying face videos based on detecting movements of facial parts, e.g., eye blinking and lip movements. These methods are suitable for static attacks, but not dynamic attacks such as replay or mask attacks. The third is image quality and reflectance-based methods, which design features to capture the superimposed illumination and noise information to the spoof images.

Most of the prior face PAD works apply SVM on hand-crafted features. While convolutional neural network (CNN) exhibits a superior performance in many computer vision tasks [6, 31, 32], there are only a few CNN-based methods for face PAD. Those methods typically use CNN for learning the representations, which will be further classified by SVM [33, 42]. In our view, further utilizing CNN in multiple ways, such as end-to-end training and learning with additional supervision, is a viable option for solving face PAD problems. On the one hand, with an increasing variety of sensing environments and PAs, it is not desirable to have a hand-crafted feature to cover all attacks. On the other hand, we need CNN to learn a robust feature from the data. With the growing numbers of face spoofing databases, CNN is known to be able to leverage the larger amount of training data and learn generalizable information to discriminate live versus spoof samples.

Following this perspective, in this chapter, we introduce a novel two-stream CNN-based face PAD method for print and replay attacks, denoted as CNN v1. The proposed method extracts the patch-based features and holistic depth maps from face images, as shown in Fig. 8.1. Here, the patch-based features are extracted from a local region of the face images, aiming at learning the spoofing texture that exists all over the images. The depth map leverages the whole face and describes the live face as a 3D object but the printed and digital screen face as a flat plain. Combining the patch-based and holistic features has two benefits: First, utilizing the local patches help to learn spoof patterns independent of spatial face areas. Second, holistic depth maps leverage the physical properties of the spoof attacks and learn a pixel-wise labeling. We use two CNNs to learn patch-based and holistic features, respectively. The first CNN is trained to predict a score for each extracted patch from a face image, and we assign the face image with the average of scores. The second CNN estimates the depth map of the face image and provides the face image with a liveness score

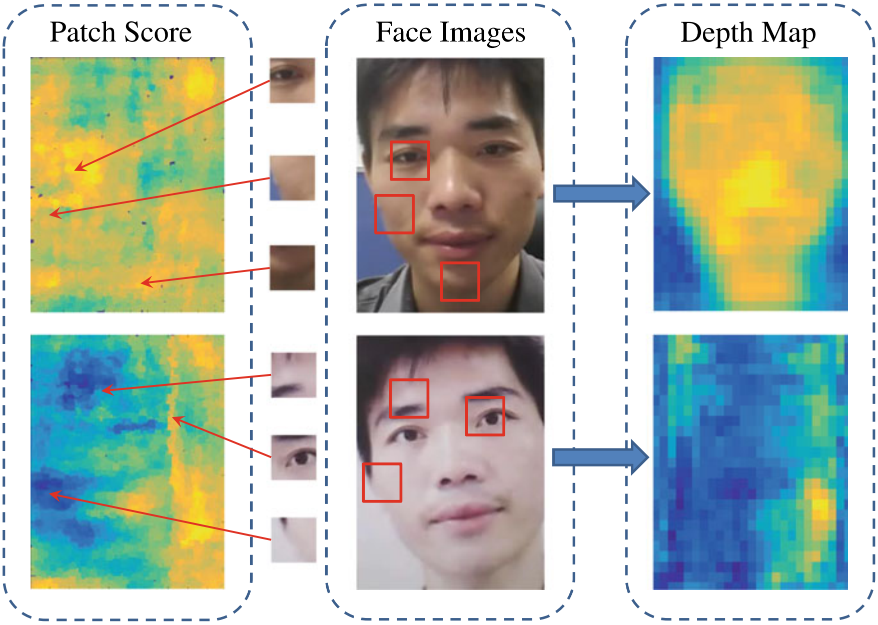


Fig. 8.1 In order to differentiate between live from spoof images, we propose an approach fusing patch-based and holistic depth-based cues. Left column shows the output scores of the local patches for a live image (top) and a spoof image (bottom), where the blue/yellow represents a high/low probability of spoof. While this visualization utilizes densely sampled patches, 10 random patches are sufficient for our anti-spoof classification. Right column shows the output of holistic depth estimation, where the yellow/blue represents closer/further points

based on estimated depth map. The fusion of the scores of both parts leads to the final estimated class of live versus spoof. The combination of these patch-based and depth-based CNNs is referred to as CNN v1. Further, to embed such PAD method in a mobile scenario, we apply an architecture optimization, a model compression, and a strategy of continuous updating. We call the advanced model as the two-stream CNN v2. The CNN v2 is trained in an end-to-end fashion and obtains comparable or higher accuracy in comparison with CNN v1, while achieving a real-time efficiency on the mobile phone system.

We summarize the contributions of this chapter as follows:

- Our proposed method utilizes both learned local and holistic features for classifying live versus spoof face samples;
- We propose a method for estimating the dense depth map for a live or spoof face image;
- We achieve the state-of-the-art performance on conventional face anti-spoofing databases;
- We provide an practical approach to train a robust and efficient system for mobile PAD scenarios.

8.2 Prior Work

We review papers in three relevant areas: traditional face PAD methods, CNN-based PAD methods, and image depth estimation.

Traditional face PAD methods Most prior work utilizes hand-crafted features and adopts shallow learning techniques (e.g., SVM and LDA) to develop a PAD system. A great number of works pay attention to the texture differences between the live faces and the spoof ones. Common local features that have been used in prior work include LBP [18, 19, 38], HOG [30, 58], DoG [44, 53], SIFT [41], and SURF [8]. However, the aforementioned features to detect texture difference could be very sensitive to different illuminations, camera devices, and specific identities. Researchers also seek solutions on different color spaces such as HSV and YCbCr [7, 10], Fourier spectra [37], and optical flow maps (OFM) [4].

Additionally, some approaches attempt to leverage the spontaneous face motions. Eye blinking is one cue proposed in [40, 52], to detect spoof attacks such as paper attack. In [29], Kollreider et al. use lip motion to monitor the face liveness. Methods proposed in [14, 15] combine audio and visual cues to verify the face liveness.

CNN-based methods CNNs have been proven to successfully outperform other learning paradigms in many computer vision tasks [6, 31, 32]. In [33, 42], the CNN serves as a feature extractor. Both methods fine-tune their network from a pretrained model (CaffeNet in [42], VGG-face model in [33]) and extract the features to distinguish live versus spoof. In [59], Yang et al. propose to learn a CNN as a classifier for face PAD. Registered face images with different spatial scales are stacked as input, and live/spoof labeling is assigned as the output. In addition, Feng et al. [20] propose to use multiple cues as the CNN input for live/spoof classification. They select shearlet-based features to measure the image quality and the OFM of the face area as well as the whole scene area. And in [57], Xu et al. propose an LSTM-CNN architecture to conduct a joint prediction for multiple frames of a video.

However, compared to other face-related problems, such as face recognition [32, 36, 55] and face alignment [26], there are still substantially fewer efforts and exploration on face PAD using deep learning techniques [3, 27, 34]. Therefore, the proposed method aims to further explore the capability of CNN in face PAD, from the novel perspective of fusing the local texture-based decision and holistic depth maps.

Image depth estimation Estimating depth from a single RGB image is a fundamental problem in computer vision. In recent years, there has been rapid progress due to data-driven methods [28], especially deep neural networks trained on large RGB-D datasets [50], as well as weak annotations [12]. Specifically, for face images, face reconstruction from one image [24, 26, 54] or multiple images [46, 47] can also be viewed as one approach for depth estimation. However, to the best of our knowledge, no prior work has attempted to estimate the depth for a spoof image, such as a face on a printed paper. In contrast, our approach estimates depth for both the live face and spoof face, which is particularly challenging since the CNN needs to discern the subtle difference between two cases in order to correctly infer the depth.

8.3 Robust CNN System for Mobile PAD

In this section, we present the details of the proposed CNN system for Mobile PAD. We first introduce a general CNN system denoted as CNN v1, which leverages two streams of CNNs: patch-based CNN and depth-based CNN. To tailor the system for a mobile scenario, we redesign the patch-based CNN and combine it with the depth-based CNN, denoted as CNN v2. In addition, we propose a simple but effective learning strategy of continuous updating to improve the robustness of the system.

8.3.1 Patch- and Depth-Based CNN v1

The proposed CNN v1 [3] consists of two streams: patch-based CNN and depth-based CNN. Figure 8.2 shows a high-level illustration of both streams along with a fusion strategy for combining them. For the patch-based CNN stream, we train a deep neural network end-to-end to learn rich appearance features, which are capable of discriminating between live and spoof face images using patches randomly extracted from face images. For the depth-based CNN stream, we train a fully convolutional network (FCN) to estimate the depth of a face image, by assuming that a print or replay presentation attack has a flat depth map, while live faces contain a normal face depth.

Either the appearance or the depth cue can detect face attacks independently. However, fusing both cues has proven to provide promising results. In this model, we refer to the fusion output as the spoof score. A face image or video clip is classified as spoof if its spoof score is above a pre-defined threshold. In the remainder of this section, we explain in detail the two CNN steams used for face PAD.

8.3.1.1 Patch-Based CNN

There are multiple motivations to use patches instead of full face in our CNN. First is to increase the number of training samples for CNN learning. Note that for all available anti-spoofing datasets, only a limited number of samples are available for

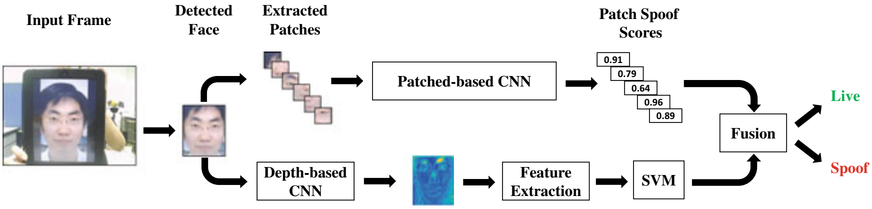


Fig. 8.2 Architecture of the proposed face PAD approach

training. For example, CASIA-FASD only contains 20 training subjects, with 12 videos per subject. Even though hundreds of faces can be extracted from each video, overfitting could be a major issue when learning the CNN due to the high similarities across the frames. Second, when using the full face images as input, traditional CNN needs to resize faces due to varying face image resolutions, where such scaling change might lead to the reduction of the discriminative information. In contrast, using the local patches can maintain the native resolution of the original face images, and thus preserve the discriminative ability. Third, assuming the spoof-specific discriminative information is present spatially in the entire face region, and patch-level input can enforce CNN to discover such information, regardless of the patch location. This is a more constrained or challenging learning task compared to using the whole face image.

Input features CNN is claimed to be a powerful feature learner that is able to map from raw *RGB* pixel intensities to the discriminative feature representation, guided by the loss function, which is in sharp difference to the conventional hand-crafted features. In our work, one observation is that CNN might also benefit from the hand-crafted features, which are proven to work well for the anti-spoof application. In a way, this is one form of bringing domain knowledge to CNN learning. This might be especially important for face anti-spoof applications, since without domain knowledge it is more likely for CNN to learn non-generalizable information from the data, rather than the true discriminative feature.

In reviewing hand-crafted features for face PAD, researchers have been experimenting with several color spaces as input to a feature extraction module to find discriminative descriptors. Typically, the most common color spaces used are *RGB*, *HSV*, YC_bC_r, and several combinations among them, such as $HSV + YC_bC_r$ [10]. The *RGB* has limited applications in face PAD due to the high correlation between the three color components and the imperfect separation of the luminance and chrominance information. On the other hand, *HSV* and YC_bC_r are based on the separation of the luminance and the chrominance information, providing additional features for learning the discriminative cues.

In this work, we attempt to use both *HSV* and YC_bC_r color spaces in the CNN-based methods. Moreover, we also explore several other input feature maps to the CNN including a pixel-wise *LBP* map and high-frequency patches. For the pixel-wise *LBP* map, we use the $LBP_{8,1}$ operator (i.e., $P = 8$ and $R = 1$) to extract the pixel-wise textural features from the face image, and afterward we randomly extract patches from the texture map. Note that in previous works, *LBP* is only used to extract histogram descriptors. For the high-frequency patches, the idea is to remove the low-frequency information from the patches which is motivated by the work in [17]. For any given face image $\mathbf{I}$, we subtract the low-pass filtered image of $\mathbf{I}$, which results in a high-frequency image $\mathbf{I}_H = \mathbf{I} - f_{lp}(\mathbf{I})$. An illustration of the various input features explored in our system is in Fig. 8.3. Compared to using RGB alone, providing these input features can facilitate the CNN training.

Based on our experiments, all of the proposed input features are useful representations to learn a CNN capable of distinguishing spoof attacks from live faces. In the experiments section, quantitative results comparing the input features will be

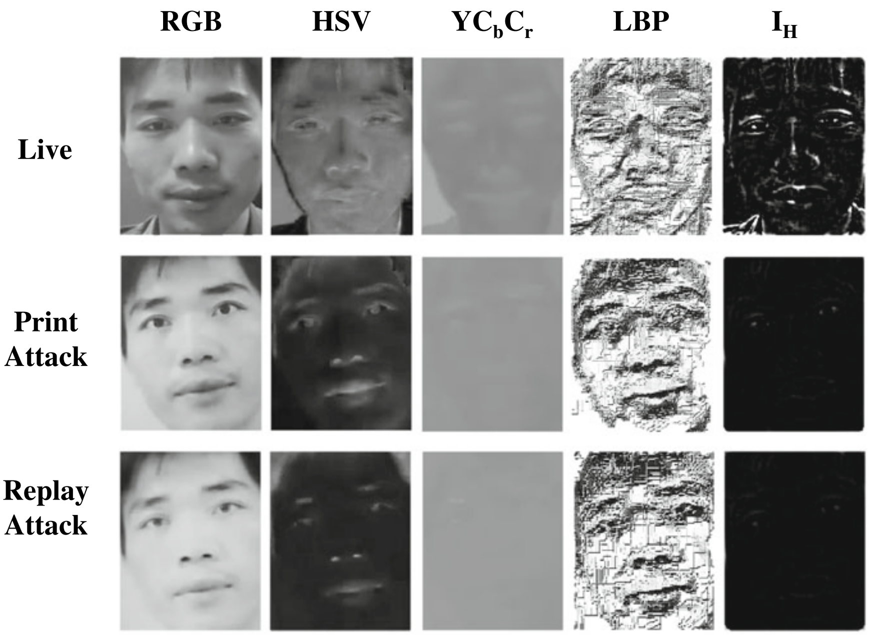


Fig. 8.3 Examples on *RGB* (G channel), *HSV* (S channel), YC_bC_r (C_b channel), pixel-wise *LBP* (*LBP* of S channel in *HSV*), high-frequency images (using G in *RGB*) of both live and spoof face images

presented. For the patch-based CNN, after detecting the face region, we convert the full face image into one of the feature representations, i.e., *HSV*, and then extract fixed size patches for CNN training and testing.

8.3.1.2 Depth-Based CNN

In this section, we explain the details of the depth-based CNN. Other than 3D-mask PA, all known PAs, such as printed paper and display, have an obviously different depth compared to the live faces. Therefore, developing a robust depth estimator can benefit the face PAD.

Based on [17], we believe that high-frequency information of face images is crucial for face PAD, and resizing images may lead to a loss of high-frequency information. Therefore, to be able to handle face images with different sizes, we propose to maintain the original image size in training the CNN for depth estimation. That is, we train a fully convolutional network (FCN) whose parameters are independent to the size of input face images. The input is face images, and the output is the corresponding depth maps. For the live faces, the depth information is from the 3D face shapes estimated using a state-of-the-art 3D face model fitting algorithm

[24–26, 35]. For the spoof faces, the depth information is the flat plain, as assumed by the attack medium's geometry, e.g., screen, paper.

Generating the depth labels We represent the live face with the dense 3D shape **A** as $\begin{pmatrix} x_1 & x_2 & \cdots & x_Q \\ y_1 & y_2 & \cdots & y_Q \\ z_1 & z_2 & \cdots & z_Q \end{pmatrix}$ where z denotes the depth information of the face, and Q is the number of 3D vertices.

Given the face image, the 3D face model fitting algorithm [24] can estimate the shape parameters $\mathbf{p} \in \mathbb{R}^{1\times 228}$ and projection matrix $\mathbf{m} \in \mathbb{R}^{3\times 4}$. We then use 3DMM model [5] to compute the dense 3D face shape **A** by

$$\mathbf{A} = \mathbf{m} \cdot \begin{bmatrix} \bar{\mathbf{S}} + \sum_{i=1}^{228} p^i \mathbf{S}^i \\ \mathbf{1}^\intercal \end{bmatrix}, \tag{8.1}$$

where $\bar{\mathbf{S}}$ is the mean shape of the face, and $\mathbf{S}^i$ are the PCA shape bases representing identification variations, e.g., tall/short, light/heavy, and expression variations, e.g., mouth opening, smile.

After we compute the 3D dense shape of the face, the depth map composes of the z-value for Q vertices from the shape **A**. In order to obtain a smoothing and consistent depth map from discrete z-values from Q vertices, the z-buffering algorithm [39] is applied, and the "texture" of the objects is imported as the depth information (i.e., z-values). To note that, input faces with different sizes would lead to a different range for z-values, mostly proportional to the face size. Hence, the depth map **M** needs to be normalized before being used as the label for CNN training. In our case, we use the max-min method for normalization.

Examples of depth maps are shown in Fig. 8.4. For spoof faces as well as the background area in the live faces, the z-value is equal to 0. Note that for some print attacks, it is possible that the papers are bent. Since it is hard to estimate the actual amount of bending, we also treat the ground truth depth of bending papers as the flat plain.

Depth map for classification The proposed FCN can estimate a depth map for a face image. Since the depth maps used to supervise the training can distinguish between live and spoof images, the estimated depth maps should also have the capability to classify live versus spoof. To leverage this capability, we train SVM classifiers using the estimated depth maps of the training data.

Specifically, to ensure that the input dimension of SVM is of the same size, the depth map **M** is overlaid with a fixed $N \times N$ grid of cells. We compute a mean depth of each local cell and generate a N^2-dim vector, which is fed to the SVM with RBF kernel. Given that resizing the depth map might lose information, we propose to train multiple SVMs with different sizes of N. To properly determine the number of SVMs, we adopt a Gaussian mixture model to fit the distribution of input image sizes. During the testing stage, we feed the testing sample to the SVM, whose input size N is closest to the sample.

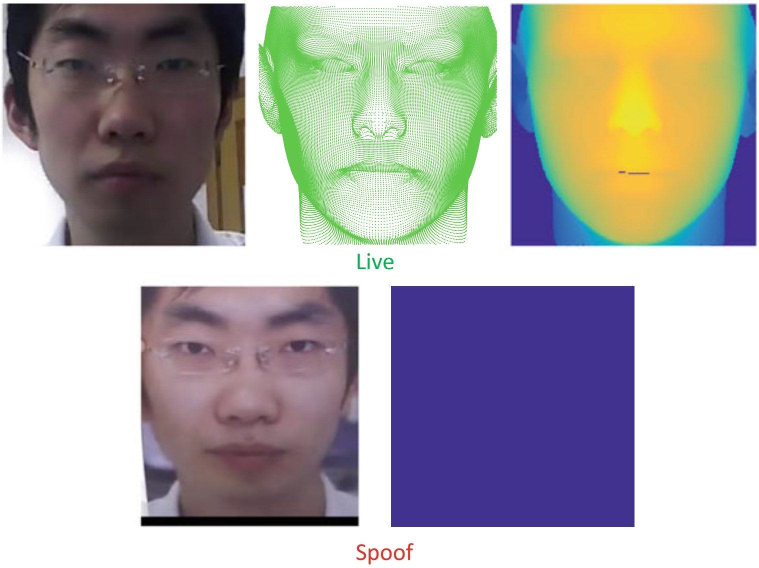


Fig. 8.4 Depth labels for depth-based CNN learning. A live face image, a fitted face model, and the depth label (top row). A spoof face image and the flat plain depth (bottom row)

Moreover, we can leverage the temporal information given a face video input. For live videos, the depth changes little over time, while the depth of spoof ones can change substantially due to noisy estimation and involuntary hand movement while holding spoof mediums. Hence for a video, we first compute a N^2-dim vector for each frame, and then compute standard deviation of the estimated depth maps of the video. The final feature of a frame feeding to SVM is a $2N^2$-dim vector. Given the SVM output of all frames, we use their average as the final score of the video.

8.3.1.3 CNN Architecture

A detailed network structure of the patch- and depth-based CNN v1 is illustrated in Table 8.1.

Patch-based CNN A total of five convolutional layers are used followed by three fully connected layers. Following every convolutional layer, we use a batch normalization, ReLU, and pooling layers. Softmax loss is utilized in CNN training. Given a training image, we initially detect the face and then crop the face region based on eye positions. After that, several patches are extracted randomly from the face image, such that all patches have the same fixed size. We avoid any rescaling to the original face images for the purpose of maintaining the spoof patterns within

Table 8.1 **a** Network structure of patch-based CNN and depth-based CNN. Red texts represent the output of the CNNs. Every convolution layer is cascaded with a ReLU layer. Note that the input size for patch-based CNN is fixed to be 96 × 96. The input size for depth-based CNN is varied from sample to sample. For simplicity, we show the case when the input size is 128 × 128. **b** The network structure of patch- and depth-based CNN v2. Red texts represent the output of the CNN. Every convolution layer is cascaded with a batch normalization and ReLU layer. Note that the input face image for the CNN v2 is normalized to 256 × 256

(*a*)

Patch-based CNN			*Depth-based CNN*		
Layer	Filter/Stride	Output Size	Layer	Filter/Stride	Output Size
			Conv-11	$3 \times 3/1$	$128 \times 128 \times 64$
Conv-1	$5 \times 5/1$	$96 \times 96 \times 50$	Conv-12	$3 \times 3/1$	$128 \times 128 \times 64$
BN-1		$96 \times 96 \times 50$	Conv-13	$3 \times 3/1$	$128 \times 128 \times 128$
MaxPooling-1	$2 \times 2/2$	$48 \times 48 \times 50$	MaxPooling-1	$2 \times 2/2$	$64 \times 64 \times 128$
			Conv21	$3 \times 3/1$	$64 \times 64 \times 128$
Conv-2	$3 \times 3/1$	$48 \times 48 \times 100$	Conv-22	$3 \times 3/1$	$64 \times 64 \times 256$
BN-2		$48 \times 48 \times 100$	Conv-23	$3 \times 3/1$	$64 \times 64 \times 160$
MaxPooling-2	$2 \times 2/2$	$24 \times 24 \times 100$	MaxPooling-2	$2 \times 2/2$	$32 \times 32 \times 160$
Conv-3	$3 \times 3/1$	$24 \times 24 \times 150$			
BN-3		$24 \times 24 \times 150$	Conv-31	$3 \times 3/1$	$32 \times 32 \times 128$
MaxPooling-3	$3 \times 3/2$	$12 \times 12 \times 150$	ConvT-32	$6 \times 6/1$	$37 \times 37 \times 128$
Conv-4	$3 \times 3/1$	$12 \times 12 \times 200$			
BN-4		$12 \times 12 \times 200$	Conv-41	$3 \times 3/1$	$37 \times 37 \times 128$
MaxPooling-4	$2 \times 2/2$	$6 \times 6 \times 200$	ConvT-42	$6 \times 6/1$	$42 \times 42 \times 128$
Conv-5	$3 \times 3/1$	$6 \times 6 \times 250$			
BN-5		$6 \times 6 \times 250$	Conv-51	$3 \times 3/1$	$42 \times 42 \times 160$
MaxPooling-5	$2 \times 2/2$	$3 \times 3 \times 250$	ConvT-52	$6 \times 6/1$	$47 \times 47 \times 160$
FC-1	$3 \times 3/1$	$1 \times 1 \times 1000$			
BN-6		$1 \times 1 \times 1000$	Conv-61	$3 \times 3/1$	$47 \times 47 \times 320$
Dropout	0.5	$1 \times 1 \times 1000$	ConvT-62	$6 \times 6/1$	$52 \times 52 \times 320$
FC-2	$1 \times 1/1$	$1 \times 1 \times 400$			
BN-7		$1 \times 1 \times 400$			
FC-3	$1 \times 1/1$	$1 \times 1 \times 2$	Conv-71	$3 \times 3/1$	$52 \times 52 \times 1$

(*b*)

Patch and Depth-based CNN v2		
Layer	Filter/Stride	Output Size
Conv-0	$3 \times 3/1$	$256 \times 256 \times 32$
MaxPooling-0	$3 \times 3/2$	$128 \times 128 \times 32$
Conv-1	$3 \times 3/1$	$128 \times 128 \times 32$
Conv-2	$3 \times 3/1$	$128 \times 128 \times 25$
Conv-3	$3 \times 3/1$	$128 \times 128 \times 32$
MaxPooling-1	$3 \times 3/2$	$64 \times 64 \times 32$
Conv-4	$3 \times 3/1$	$64 \times 64 \times 32$
Conv-5	$3 \times 3/1$	$64 \times 64 \times 25$
Conv-6	$3 \times 3/1$	$64 \times 64 \times 32$
MaxPooling-2	$3 \times 3/2$	$32 \times 32 \times 32$
Conv-7	$3 \times 3/1$	$32 \times 32 \times 32$
Conv-8	$3 \times 3/1$	$32 \times 32 \times 25$
Conv-9	$3 \times 3/1$	$32 \times 32 \times 32$
Concat	MaxPooling-1 + MaxPooling-2 + Conv-9	
Conv-10	$3 \times 3/1$	$32 \times 32 \times 32$
Conv-11	$3 \times 3/1$	$32 \times 32 \times 25$
Conv-12	$3 \times 3/1$	$32 \times 32 \times 2$

the extracted patches. If the face image is a live face, we assign all of its patches a binary label of 1. If the face is a spoof face, the labels of patches are 0.

During testing, we extract patches in the same manner as training. The patch-based CNN will produce spoof scores for every patch in the range of 0–1. The final result of the image is the average spoof score of all patches. If the presentation attack is in the video format, we compute the average spoof score across all frames.

Depth-based CNN We employ a FCN to learn the nonlinear mapping function $f(\mathbf{I}; \Theta)$ from an input image $\mathbf{I}$ to the corresponding depth map $\mathbf{M}$, where Θ is the network parameter. Following the setting in Sect. 8.3.1.1, we use $HSV + YC_bC_r$ features as the CNN input. The depth label $\mathbf{M}$ is obtained in the approach described in the previous subsection. Our FCN network has a bottleneck structure, which contains two parts, downsampling part and upsampling part, as shown in Table 8.1. The downsampling part contains six convolution layers and two max-pooling layers; the upsampling part consists of $five$ convolution layers which sandwich $four$ transpose convolution layers for the upsampling purpose. This architecture composes of only convolution layers without fully connected layer, and each layer is followed by the ReLU layer. We define the loss function as the pixel-level Euclidean loss,

$$\arg\min_{\Theta} J = \|f(\mathbf{I}; \Theta) - \mathbf{M}\|_F^2 . \tag{8.2}$$

8.3.2 *Patch- and Depth-Based CNN v2*

As we evaluate the patch- and depth-based CNN v1, we notice several drawbacks of this model. First, it is not very time-efficient. With N random patches and one whole image to go through the CNNs for each sample, the system requires approximately $1 + \frac{N}{2}$ seconds to process each frame, which is not suitable for mobile applications such as phone unlocking. To reduce the time cost, we revisit the patch-based approach and propose a fully convolution network that learns the same patch-based features. Secondly, CNN v1 is trained with limited amount of data, where most of them are captured in a constrained environment. It could make wrong estimations for input samples with extreme illuminations, poses, and other unknown factors. To handle this real-world issue, we deploy a simple but effective strategy of continuous updating that can improve the generalization of the system by a large margin. The overall architecture of the CNN v2 is shown in Fig. 8.5.

8.3.2.1 Revisit the Patch-Based CNN

We mention several motivations to use patches instead of the full face in CNN in Sect. 8.3.1.1, and one of the major reasons is to prevent overfitting of the CNN training. However, during the testing time, we need to sample sufficient amount of random patches in order to maintain a consistent performance, which can be very time-consuming. We revisit the design of the patch-based CNN. For the framework of CNN, the effective receptive field for a certain level of convolution layer is constrained. For example, the conv-13 in Tabe 8.1 has a receptive field of 5 × 5 patch in

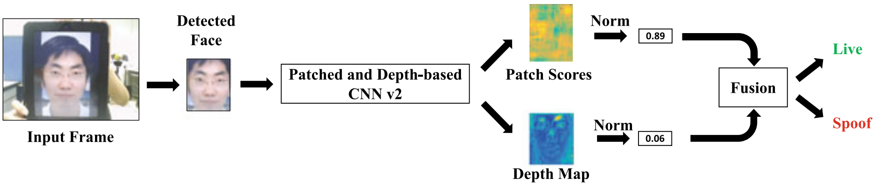


Fig. 8.5 Architecture of the advanced face PAD approach

the original input. With a specified depth, the output of a fully convolutional network (FCN) is locally receptive and hence essentially *patch-based*.

Therefore, we redesign the patch-based CNN as a fully convolutional network, where the input of the network is the cropped face, and the output of the network is a binary mask. Each element of the mask indicates the spoofness of a local patch whose size is the same as that of the receptive field. For a spoof face, by assuming the whole region of the face as well as its close surrounding are all from the spoof medium, such as printed paper and digital screen, the binary mask is an *one* map. And for a live face, the binary mask is a *zero* map.

Despite the patch-based CNN shares similar architecture as the depth-based CNN, they still learn distinct features for detecting PA. The patch-based CNN focuses on generic spoofing features that exist on all local regions of the image, while the depth-based CNN focuses on face-dependent features that only exist within face regions, such as eye corners, cheek, and jaw lines.

8.3.2.2 Maps for Classification

Patch-based CNN provides a binary mask map to indicate the spoofness of each local patch, and depth-based CNN provides an estimated depth map for the whole face region. Since the given labeling (i.e., binary mask or depth map) itself is discriminative with respect to live versus spoof, it might not be necessary to use an additional classifier (e.g., SVM) for classification. To convert these two maps into a score for decision making, we simply compute the L_2 norm of each map and sum them up with an assigned weight, as shown in Eq. 8.3.

$$score = \alpha \left\|\mathbf{M}\right\|_2^2 + \left\|\mathbf{D}\right\|_2^2 . \tag{8.3}$$

8.3.2.3 Model Compression

To utilize the trained face PAD CNN model for authentication in mobile and embedded devices, we should make the CNN model compatible with the computational power on those devices. There are many research papers for compressing the CNN models and reducing their computational cost. The model compressing methods [13] can be categorized into four main groups: (1) parameter pruning and sharing [51], (2) knowledge distillation [21], (3) transferred convolutional filters [48], (4) low-rank factorization [22].

In this work, our objective is to find the model with the minimum computational requirement; hence, we design the model compression as a search algorithm. We utilize a new greedy method similar to the binary search for finding the minimum number of filters needed in each layer. We make a development set for evaluating the performance of the compressed models with our greedy method. To find the minimum size of the network with acceptable performance on the development set, we iteratively reduce the number of filters by half while keeping the number of layers fixed and retraining the network. We stop this process when the CNN model cannot achieve acceptable performance on the development set, as an indication of low capacity of the CNN model for the face anti-spoofing task. By applying this method, we reduce the size of the CNN model by 160 times from $\sim$ 80 Mb to 0.5 Mb while achieving similar performance.

8.3.2.4 CNN v2 Architecture

A detailed network structure of the patch- and depth-based CNN v2 is illustrated in Table 8.1. With the same input to the network, the new patch-based CNN and depth-based CNN can share the weights with each other, since they are trained for the same purpose. We combine these two networks into one to further reduce the computation time by half. After model compression, a total of nine convolution layers with three max-pooling layers are used to extract spoofing features at different levels of scales. Then, we adopt a shortcut connection to concatenate the feature maps of each pooling layer with the size normalized as 32. The concatenated features are sent to three additional convolution layers. The final output of the network is two 32×32 maps, where the first map is supervised by the zero/one map for learning the patch-based features, and the second map is supervised by the depth map for learning the depth-based face-dependent features. The maximum receptive field of the CNN v2 is 72. The L_1 loss is utilized in CNN training. Following every convolution layer, we use a batch normalization layer and ReLU layer.

Given a testing image, we initially detect the face and then crop the face region based on eye positions. The cropped face is utilized in the CNN v2 to produce the binary mask map $\mathbf{M}$ as well as the depth map $\mathbf{D}$. The final score for the testing sample is a weighted average of the map norms, as shown in Eq. 8.3.

8.3.2.5 CNN System Updating

To improve performance and increase the robustness of the network, we utilize iterative update training by incorporating failure cases from the previous model. This process is shown in Fig. 8.6. We begin with a trained model and its corresponding set of training data. The performance of the trained model is qualitatively analyzed by collecting failure cases using our PC and Android demo applications. These failure cases are then analyzed, specifically considering if there are patterns that are common to the experienced failures such as low illumination, reflective glare on spoofs, washing out due to excessive illumination, and extreme pose, among others. The collected failure case images are then added into the training data, and the model is updated using a random shuffle of the previous and newly collected data. In this way,

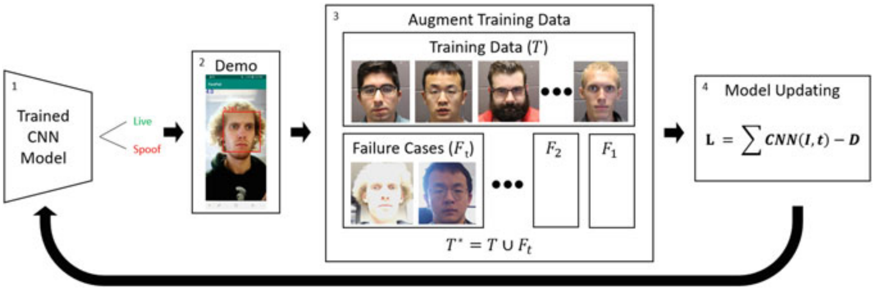


Fig. 8.6 Iterative updating of the trained model using the most recent failure cases collected by the PC and Android apps allows for targeted improvements for situations in which the model fails. The updating process begins with a model trained on our training dataset. Newly collected data is added to the current training data. This significantly and quickly improves the trained model without unnecessary effort to collect unimportant data

we enforce that the model performs similarly on previous success cases and previous failure cases, while improving performance on the most recent failure cases. As we repeat the updating process multiple times, it becomes more difficult to collect failure cases, indicating that the model has become more robust to its previous weaknesses.

8.4 Experiments

8.4.1 Database

We evaluate our proposed method on two PAs: print and replay attacks, using five benchmark databases: CASIA-MFSD [60], MSU-USSA [41], replay attack [16], OULU-NPU [11], and SiW [34].

CASIA-MFSD: This database contains 50 subjects and 12 videos for each subject under $three$ different image resolutions and varied lightings. Each subject includes $three$ different spoof attacks: replay, warp print, and cut print attacks. Due to the diversity of the spoof types, many previous works [40, 52] that leverage the motion cues such as eye blinking or shape deformation would fail on this dataset. This dataset partitions the subject space and uses 20 subjects for training and 30 subjects for testing.

MSU-USSA: As one of the largest public face spoofing databases, MSU-USSA contains 1000 in the wild live subject images from the weakly labeled face Database and creates $eight$ types of spoof attacks from different devices such as smart phones, personal computers, tablets, and printed papers. This dataset covers images under different illuminations, image qualities, and subject diversity.

Replay attack: This database contains $1,300$ live and spoof videos from 50 subject. These videos are divided into training, development, and testing sets with 15, 15, and 20 subjects, respectively. The videos contain two illumination conditions: controlled and adverse. Given the print and replay attacks in this set, the database also divides the attacks into two more types based on whether they use a support to hold the spoof medium, or if the attack is held by a person.

OULU-NPU: This more recent database is comprised of 4920 live and spoof videos captured of 55 subjects using six mobile phone cameras in $three$ sessions with varying illumination conditions and scene backgrounds. Unlike earlier databases, this uses $1080p$ videos to accommodate higher quality images' increasing prevalence in society. Four testing protocols are defined to evaluate a network's performance under differing situations such as generalization in leave-one-out testing.

8.4.2 Experimental Parameters and Setup

In CNN v1, we use Caffe toolbox [23] to implement the patch-based CNN. The learning rate is set as 0.001, decay rate as 0.0001, momentum as 0.99, and batch size as 100. Before being fed into the CNN, the face samples are normalized by subtracting the mean face of training data. Since CASIA and replay attack are video datasets, we only extract two random patches per frame for training. For the images in MSU-USSA, we extract 64 patches from each live face region and $eight$ patches from each spoof face region. For CASIA and MSU-USSA, a fixed patch size of 96×96 is used. For replay attack, given its low image resolution, the patch size is 24×24. To accommodate the difference in patch sizes, we remove the first two pooling layers for the patch-based CNN. For the depth-based CNN, we use TensorFlow [1], with the learning rate of 0.01 and batch size of 32. The patches are also normalized by subtracting the mean face of training data. When generating the depth labels, we normalize the depth in the range of 0–1. We use the weighted average of two streams' scores as the final score of our proposed method, where the weights are experimentally determined.

In CNN v2, we use TensorFlow to implement all parts, with the learning rate of 0.01 and batch size of 32. The input face samples are normalized to be $256 \times 256 \times 3$. The depth maps are also normalized to the range of 0–1 with the size of 32. We use the weighted average of two feature maps as the final score as mentioned in Sect. 8.3.2.2. Based on the experiments, the final α and β are set to be 0.5 and -1.2, respectively, for all of the following experiments.

Our experiments follow the protocol associated with each of the five databases. For each database, we use the training set to learn the CNN models and the testing set for evaluation in terms of equal error rate (EER) and half total error rate (HTER).

8.4.3 Ablation Study

8.4.3.1 Patch-Based CNN Analysis

In Sect. 8.3.1.1, we explore several input feature maps to train the patch-based CNN v1, which include different combinations of color spaces, a pixel-wise *LBP* map, and high-frequency patches. For all of the experiments, we first detect and then crop the face for a given frame. After that we convert the face image into a new feature map as seen in Fig. 8.3, which will then be used to extract patches. Table 8.2 presents the results on CASIA-FASD when using different combinations of input feature maps. Based on our experiments, we only show the best four combinations of features in this table. From these results, we can clearly see that the $HSV + YC_bC_r$ features have a significant improvement in performance compared to the other features with an EER of 4.44% and an HTER of 3.78%. Moreover, adding an *LBP* map to the $HSV + YC_bC_r$ has a negative impact to the CNN learning, which reduces the performance of using $HSV + YC_bC_r$ only by 2.31% HTER. Similarly, when training the patch-based CNN with high-frequency data in the $HSV + YC_bC_r$ images, it also reduces the performance by 1.79% HTER. This shows that the low frequencies may also provide discriminative information to anti-spoofing.

8.4.3.2 Depth-Based CNN Analysis

The depth map results of CNN v1 on the CASIA-FASD testing set are shown in Fig. 8.7. The CNN is attempting to predict the face-like depth of a live face, i.e., higher values in the depth map, while the predicted depth of the spoof images to be flat, i.e., lower values in the depth map. Due to the difficulty of the problem, the estimated depth is not perfect, compared to the depth label shown in Fig. 8.4. However, we can still find a clear distinction between the depth maps of the live images and those of the spoof images. In the spoof image, there might be certain areas that suffer more degradation and noise from the spoof attack. As we can see from Fig. 8.7, our CNN is still trying to predict some areas with high values in the depth map. However, overall depth patterns of spoof samples are far from those of live samples so that the SVM can learn their difference. Hence, training a CNN for depth estimation is beneficial to

Table 8.2 EER (%) and HTER (%) of CASIA-FASD, when feeding different features to patch-based CNN

Feature	EER (%)	HTER (%)
YC_bC_r	4.82	3.95
$YC_bC_r + HSV$	**4.44**	**3.78**
$YC_bC_r + HSV + LBP$	7.72	6.09
$(YC_bC_r + HSV)_H$	9.58	5.57

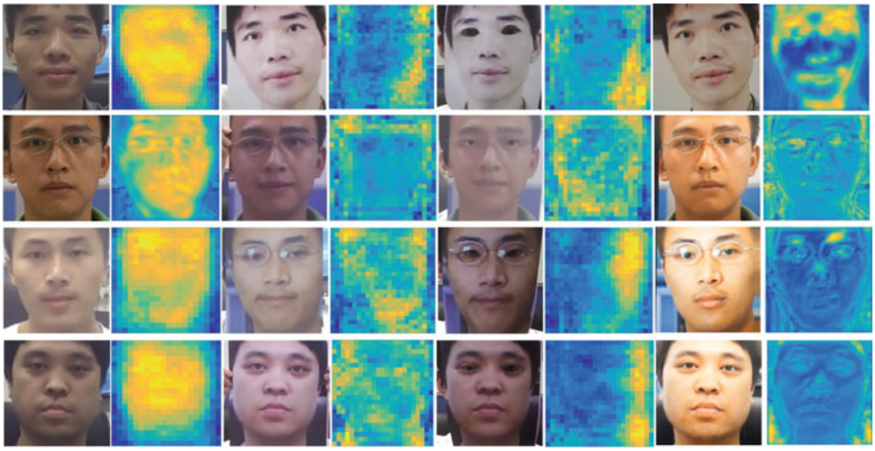


Fig. 8.7 Depth estimation on CASIA-FASD testing subjects. The first two columns are the live images and their corresponding depth maps, and the rest six columns are three different types of spoof attacks (print, cut print, and video attacks) and their corresponding depth maps

face anti-spoofing. Fig. 8.8 shows the mean and standard deviation of the estimated depth maps for all live faces and spoof faces in replay attack. The differences of live vs. spoof in both mean and standard deviation demonstrate the discriminative ability of depth maps, as well as support the motivation of feature extraction for SVM in Sect. 8.3.2.2.

8.4.3.3 Fusion Analysis

We extensively analyze the performance of our patch-based and depth-based CNN v1 on CASIA-FASD and report frame-based performance curves as seen in Fig. 8.9. As mentioned earlier, CASIA-FASD has three different video qualities and three different presentation attacks, which we use to highlight the differences of our proposed CNN streams. For the low-quality images, the patch-based method achieves an EER of 2.78%. For the same quality, we notice that depth-based CNN performs better, which is understandable since the relative depth variation of frontal-view face image is very small compared to the far distance when a low-quality face image is captured. For the normal quality, the fusion of both methods has a large positive impact on the final result, which can be seen from the ROC curves. The result of both methods on high-quality videos is reasonably good, and therefore, fusion will maintain the same performance. It is clear that the depth-based method struggles when the face images are lower in resolution, and vice-versa for the patch-based method. On the other hand, the patch-based method suffers with high resolution, and vice-versa for the depth-based method. Therefore, the fusion of both methods will strengthen the weak part of either one.

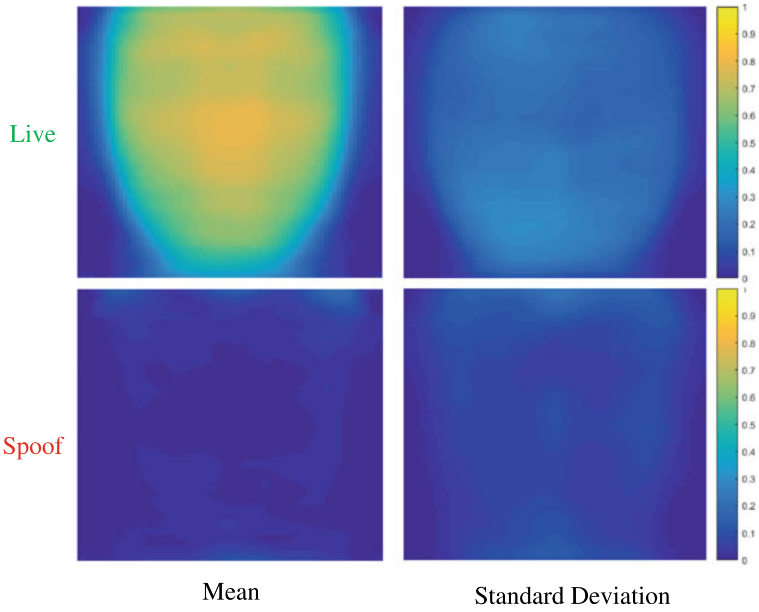


Fig. 8.8 Mean and standard deviation of the estimated depth maps of live and spoof faces, for all testing samples in replay attack. Note the clear differences in both the mean and standard deviation between the two classes

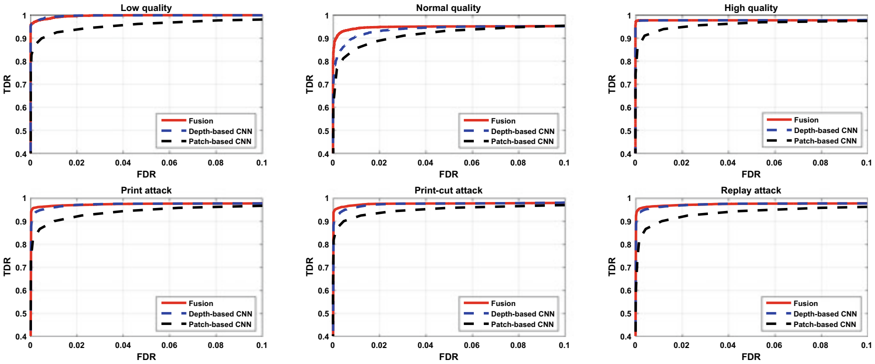


Fig. 8.9 Frame-based ROC curves on CASIA-FASD comparing the fusion method with the patch-based and depth-based CNNs

When analyzing the three different presentation attacks in CASIA-FASD with our proposed methods, the most successfully detected attack is the video replay attack. It is worthy to note that, since the ROC curve of every attack is an average of the three different video qualities, the difference among the three attacks is not large. For the fusion results, the best gain can be seen in the print attacks compared to the results of the two methods independently.

8.4.3.4 Patch- and Depth-Based CNN v2 Analysis

Similarly, we evaluate the performance of the patch- and depth-based CNN v2. In CNN v2, we utilize the CNN to estimate two maps, a binary mask of spoofness, and a depth map. We train a CNN with binary mask supervision only and a CNN with depth map supervision only to validate the effectiveness of fusing the two feature maps. The test is conveyed on CASIA-FASD database. Without the supervision of depth map, the CNN obtains 6.7% as the EER and 6.3% as the HTER; without the supervision of binary mask, the CNN obtains 25.6% as the EER and 20.4% as the HTER. By combining the two streams, the CNN v2 can achieve the best performance of 4.4% as the EER and 4.6% as the HTER.

To further show the effectiveness of the continuous updating strategy, we collect a private testing set. To continuously update the model, we use the face PAD demo system with CNN v2 to capture failure cases from $five$ subjects, none of which are included in the private testing set. The model without updating obtains 31.2% as the EER and 31.1% as the HTER, while the model with updating achieves 6.2% as the EER and 5.4% as the HTER, which demonstrates a large margin of improvement.

8.4.4 Experimental Comparison

We compare the proposed method with the state-of-the-art CNN-based methods on CASIA-FASD. Table 8.3 shows the EER and HTER of six face anti-spoof methods. Among different methods in Table 8.3, the temporal features are utilized in a long short-term memory (LSTM) CNN [57], the holistic features are extracted for classification in [59], CNN is used for the feature extraction in [33], and after applying PCA to the response of the last layer, SVM is utilized for classification. According to Table 8.3, our method outperforms others in both EER and HTER. This shows the combination of local and holistic features contain more discriminative information. Note that even though depth-based CNN alone has larger errors, its fusion with patch-based CNN still improves the overall performance. For CNN v2, it shows a perfect performance on replay database and the high-resolution part of CASIA dataset with EER and HTER to be 0. However, it performs worse on the low-resolution part of the CASIA dataset, and thus the overall EER and HTER are slightly worse than CNN v1. Because of the superior performance on the first two part and its time efficiency, we still regard CNN v2 as a better model and use CNN v2 to conduct further experiments.

Table 8.3 EER (%) and HTER (%) on CASIA-FASD

Method	EER (%)	HTER (%)
Fine-tuned VGG-face [33]	5.20	–
DPCNN [33]	4.50	–
[59]	4.92	–
CNN [57]	6.20	7.34
[10]	6.2	–
[49]	3.14	–
[8]	2.8	–
[57]	5.17	5.93
Haralick features [2]	–	1.1
Moire pattern [43]	–	0
Patch-based CNN	4.44	3.78
depth-based CNN	2.85	2.52
Patch- and depth-based CNN v1	**2.67**	**2.27**
Patch- and depth-based CNN v2	4.4	4.6

Table 8.4 EER (%) and HTER (%) on MSU-USSA

Method	EER (%)	HTER (%)
[41]	3.84	–
Patch-based CNN	0.55 ± 0.26	0.41 ± 0.32
depth-based CNN	2.62 ± 0.73	2.22 ± 0.66
Patch and depth-based CNN v1	0.35 ± 0.19	0.21 ± 0.21
Patch and depth-based CNN v2	**0** ± 0	**0** ± 0

We also test our method on the MSU-USSA database. Not many papers report results in this database because it is relatively new. Table 8.4 compares our results with [41] which analyzes the distortions in spoof images and provides a concatenated representation of LBP and color moment. In comparison with [41], our patch-based CNN already achieves 89% reduction of EER. The complementariness of depth-based CNN further reduce both the EER and HTER.

On the replay attack database [16], we compare the proposed method with three prior methods in Table 8.5. For the CNN v1, although our EER is similar to the prior methods, the HTER of our method is much smaller, which means we have fewer false acceptance and rejection. Moreover, though the fusion does not significantly reduce the EER and HTER over the depth-based CNN, we do observe an improvement in the AUC from 0.989 in patch-based CNN to 0.997 in the fusion. Additionally, our CNN v2 is able to achieve perfect performance on the replay attack dataset, demonstrating the enhanced capability of the CNN v2.

Table 8.5 EER (%) and HTER (%) on replay attack

Method	EER (%)	HTER (%)
Fine-tuned VGG-face [33]	8.40	4.30
DPCNN [33]	2.90	6.10
[59]	2.14	–
[10]	0.4	2.9
[8]	0.1	2.2
Moire pattern [43]	–	3.3
Patch-based CNN	2.50	1.25
Depth-based CNN	0.86	0.75
Patch- and Depth-based CNN v1	0.79	0.72
Patch- and Depth-based CNN v2	0	0

Table 8.6 Performance of the proposed method and SOTA face anti-spoofing methods during cross-dataset evaluation on current face anti-spoofing datasets. The HTER is reported. Results for other works are reported for cross-dataset testing between CASIA-FASD and replay attack. Results for OULU and the private test set are the HTER over the entire test set. The results for our models are only reported if the model evaluated was not trained on the corresponding training data for a given test set

Algorithm	CASIA-FASD	replay attack	OULU	Private test
Motion [19]	47.9	50.2	–	–
Spectral cubes [45]	50.0	34.4	–	–
Color LBP [10]	35.4	37.9	–	–
Color texture [7]	37.7	30.3	–	–
Color SURF [9]	**23.2**	26.9	–	–
Boulkenafet [9]	39.2	**9.6**	–	–
Liu [34]	27.6	28.4	–	–
CNN v2 (CASIA baseline)	–	42.0	25.3	26.7
CNN v2 (Replay baseline)	43.2	–	36.2	27.8
CNN v2 (Without updating)	36.1	34.7	33.1	31.1
CNN v2 (With updating)	**23.2**	**15.4**	**0.0**	**5.4**

Table 8.6 shows the performance of the CNN v2 compared to SOTA performance for cross-dataset evaluation. In this cross-dataset evaluation scenario, the HTER for all methods is significantly poorer than in the intra-dataset evaluation scenario, as is evident in our best performance in HTER of 23.2% compared to 2.3% for CASIA-FASD and 15.4% compared to 0.0% for replay attack. Our CASIA baseline and replay baseline performance are for the CNN v2 trained on CASIA-FASD and replay attack, respectively. When trained only on this low-resolution data, the cross-dataset performance is poor. The CNN v2 without updating, which was trained on a

larger dataset of mostly 1080p images, lags slightly behind the SOTA performance. However, when we incorporate the iterative updating of the network, we are able to achieve SOTA performance for cross-dataset evaluation, except in the case of the replay attack dataset. However, the SOTA work that performs best on replay attack performs much worse than our CNN v2 with updates on CASIA-FASD, indicating that our CNN v2 with updating is much more stable and able to generalize well. This further demonstrates the value of iterative updates using failure cases to improve the robustness of the network, even in the case of cross-dataset evaluation.

8.4.5 Model Size and Running Time

Figure 8.10 shows the Android demo app under three different situations. The Android demo has two major functions, testing the performance of the trained model and collecting failure cases for the current model. This is accomplished via three modes in the demo: (*i*) normal mode, (*ii*) live failure mode, and (*iii*) spoof failure mode. A prediction of live will draw a green bounding box around the face. In normal mode, the score is displayed to the screen. In live failure mode, it is assumed that any detected faces are live. A prediction of spoof will save the image to the phone's

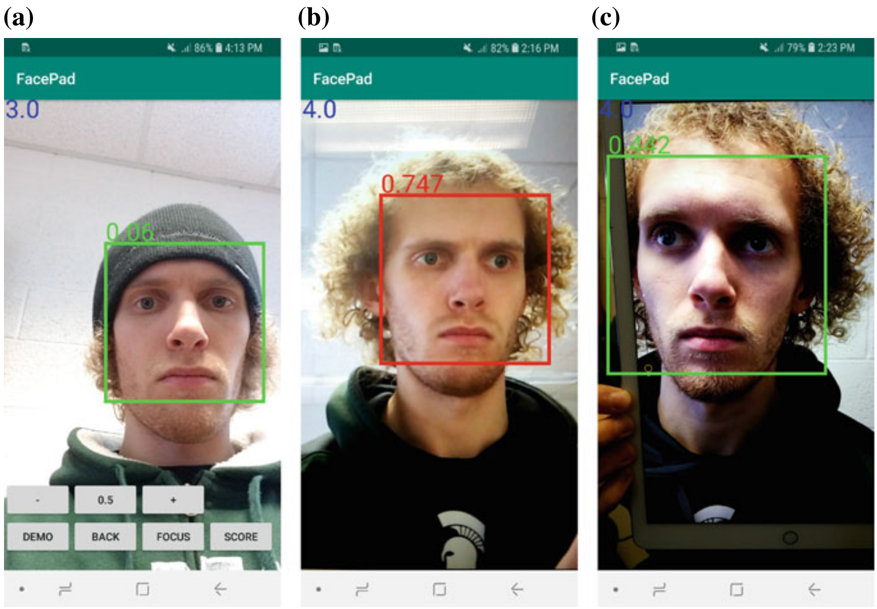


Fig. 8.10 Screenshot of the Android mobile phone demo. A score of 0 indicates live, and a score of 1 indicates spoof. Shown are **a** correct detection of a live face, **b** correct detection of a spoof attack, **c** failed detection of a spoof attack and subsequent capture of the failure case

storage as a live failure case. In spoof failure mode, it is assumed that any detected faces are spoof. A prediction of live will save the image to the phone's storage as a spoof failure case.

The demo mode and failure case threshold can be changed via the in-app settings (shown in Fig. 8.10). Modifying the failure case threshold, let us tune the difficulty for a failure case to occur by requiring higher confidence as the updating process matures. These settings also allow for hiding the raw score value, switching between the front and rear cameras of the device and refocusing the camera. Similar settings are available on the PC demo for collecting failure cases.

In an iterative method, the failure cases collected by both the PC and Android demos are added to the training set, and the model is updated with these additional images. This allows for rapid improvement of common failure cases across any of our demo-enabled devices. Often a failure case from a previous iteration will be correctly classified by the updated model, thereby reducing the number of failure cases collected each iteration. The reduced number of failure cases indicates that the updated model is becoming increasingly robust against attacks it was previously weak to. As shown in Table 8.6, the model with updating performs significantly better than the non-updated model.

Due to the limited computation ability of smartphone devices compared to PCs, we must reduce the memory and processing time of the trained model for the Android demo. To do this, we reduce the number of kernels in the convolutional layers until an appropriately small and fast, but still accurate model is produced. This improves the responsiveness of the Android app by doubling its FPS, but requires a small degradation in performance. Finally, we are able to achieve $nine$ FPS on the PC application without using the GPU, which increases to 30 FPS when we enable the GPU. We are unable to utilize the GPU on Android smartphones, and hence are limited to *four* FPS for the reduced CNN v2 model or 1–2 FPS for the full CNN v2 model.

8.5 Conclusions

In this chapter, we introduce a novel solution for mobile face PAD system via fusing patch-based CNN and depth-based CNN. Unlike the prior PAD methods that use the full face and single labels to train and detect presentation attacks, we leverage both supervisions for local patches from the same face and estimation of the face depth to distinguish the spoof from live faces. The first CNN stream is based on patch appearance extracted from face regions. This stream demonstrates its robustness across all presentation attacks, especially on lower-resolution face images. The second CNN stream is based on face depth estimation using the full face image. We prove it can improve the performance via fusing two CNNs. We further improve its mobile performance via combining two CNNs into one, optimizing the network structures and applying the strategy of continuous learning. The experiments show that the proposed CNN is robust, generalized, and computationally efficient in several testing scenarios, either intra-testing on the commonly used database, or testing sample from real-world hard cases.

References

1. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M, Kudlur M (2016) Tensorflow: a system for large-scale machine learning. In: OSDI, vol 16, pp 265–283
2. Agarwal A, Singh R, Vatsa M (2016) Face anti-spoofing using Haralick features. In: 2016 IEEE 8th international conference on biometrics theory, applications and systems (BTAS). IEEE, pp 1–6
3. Atoum Y, Liu Y, Jourabloo A, Liu X (2017) Face anti-spoofing using patch and depth-based CNNs. In: 2017 IEEE international joint conference on biometrics (IJCB). IEEE, pp 319–328
4. Bao W, Li, H, Li, N, Jiang W (2009) A liveness detection method for face recognition based on optical flow field. In: International conference on image analysis and signal processing, 2009. IASP 2009. IEEE, pp 233–236
5. Blanz V, Vetter T (2003) Face recognition based on fitting a 3D morphable model. IEEE Trans Pattern Anal Mach Intell 25(9):1063–1074
6. Blunsom P, Grefenstette E, Kalchbrenner N (2014). A convolutional neural network for modelling sentences. In Proceedings of the 52nd annual meeting of the association for computational linguistics
7. Boulkenafet Z, Komulainen J, Hadid A (2016) Face spoofing detection using colour texture analysis. IEEE Trans Inf Forensics Secur 11(8):1818–1830
8. Boulkenafet Z, Komulainen J, Hadid A (2017) Face antispoofing using speeded-up robust features and fisher vector encoding. IEEE Signal Process Lett 24(2):141–145
9. Boulkenafet Z, Komulainen J, Hadid A (2018) On the generalization of color texture-based face anti-spoofing. Image Vis Comput 77:1–9
10. Boulkenafet Z, Komulainen J, Hadid A (2015) Face anti-spoofing based on color texture analysis. In: 2015 IEEE international conference on image processing (ICIP). IEEE, pp 2636–2640
11. Boulkenafet Z, Komulainen J, Li L, Feng X, Hadid A (2017) OULU-NPU: a mobile face presentation attack database with real-world variations. In: 2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017). IEEE, pp 612–618
12. Chen W, Fu Z, Yang D, Deng J (2016) Single-image depth perception in the wild. In: Advances in neural information processing systems, pp 730–738
13. Cheng Y, Wang D, Zhou P, Zhang T (2017) A survey of model compression and acceleration for deep neural networks. arXiv:1710.09282
14. Chetty G (2010) Biometric liveness checking using multimodal fuzzy fusion. In 2010 IEEE international conference on fuzzy systems (FUZZ). IEEE, pp 1–8
15. Chetty G, Wagner M (2006) Audio-visual multimodal fusion for biometric person authentication and liveness verification. In: Proceedings of the 2005 NICTA-HCSNet multimodal user interaction workshop, vol 57. Australian Computer Society Inc., pp 17–24
16. Chingovska I, Anjos A, Marcel S (2012) On the effectiveness of local binary patterns in face anti-spoofing. In: Proceedings of the 11th international conference of the biometrics special interest group (No. EPFL-CONF-192369)
17. da Silva Pinto A, Pedrini H, Schwartz W, Rocha A (2012) Video-based face spoofing detection through visual rhythm analysis. In: 2012 25th SIBGRAPI conference on graphics, patterns and images (SIBGRAPI). IEEE, pp 221–228
18. de Freitas Pereira T, Anjos A, De Martino JM, Marcel S (2012) LBP-TOP based countermeasure against face spoofing attacks. In: Asian conference on computer vision. Springer, Berlin, Heidelberg, pp 121–132
19. de Freitas Pereira T, Anjos A, De Martino JM, Marcel S (2013) Can face anti-spoofing countermeasures work in a real world scenario? In: 2013 International conference on biometrics (ICB). IEEE, pp 1–8
20. Feng L, Po LM, Li Y, Xu X, Yuan F, Cheung TCH, Cheung KW (2016) Integration of image quality and motion cues for face anti-spoofing: a neural network approach. J Vis Commun Image Represent 38:451–460

21. Hinton G, Vinyals O, Dean J (2015) Distilling the knowledge in a neural network. arXiv:1503.02531
22. Jaderberg M, Vedaldi A, Zisserman A (2014) Speeding up convolutional neural networks with low rank expansions. arXiv:1405.3866
23. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T. (2014) Caffe: convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM international conference on Multimedia. ACM, pp 675–678
24. Jourabloo A, Liu X (2017) Pose-invariant face alignment via CNN-based dense 3D model fitting. Int J Comput Vis 124(2):187–203
25. Jourabloo A, Liu X (2015) Pose-invariant 3D face alignment. In: Proceedings of the IEEE international conference on computer vision, pp 3694–3702
26. Jourabloo A, Liu X (2016) Large-pose face alignment via CNN-based dense 3D model fitting. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4188–4196
27. Jourabloo A, Liu Y, Liu X (2018) Face de-spoofing: anti-spoofing via noise modeling. In: European conference on computer vision. Springer, Cham, pp 297–315
28. Karsch K, Liu C, Kang SB (2014) Depth transfer: depth extraction from video using non-parametric sampling. IEEE Trans Pattern Anal Mach Intell 36(11):2144–2158
29. Kollreider K, Fronthaler H, Faraj MI, Bigun J (2007) Real-time face detection and motion analysis with application in “liveness” assessment. IEEE Trans Inf Forensics Secur 2(3):548–558
30. Komulainen J, Hadid A, Pietikainen M (2013) Context based face anti-spoofing. In: 2013 IEEE sixth international conference on biometrics: theory, applications and systems (BTAS). IEEE, pp 1–8
31. Krizhevsky A, Sutskever I, Hinton GE (2012). Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
32. Lawrence S, Giles CL, Tsoi AC, Back AD (1997) Face recognition: a convolutional neural-network approach. IEEE Trans Neural Netw 8(1):98–113
33. Li L, Feng X, Boulkenafet Z, Xia Z, Li M, Hadid A (2016). An original face anti-spoofing approach using partial convolutional neural network. In: 2016 6th international conference on Image processing theory tools and applications (IPTA). IEEE, pp 1–6
34. Liu Y, Jourabloo A, Liu X (2018) Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 389–398
35. Liu Y, Jourabloo A, Ren W, Liu X (2017) Dense face alignment. In: Proceedings of IEEE international conference on computer vision workshops, pp 1619–1628
36. Liu F, Zeng D, Zhao Q, Liu X (2018) Disentangling features in 3D face shapes for joint face reconstruction and recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5216–5225
37. Li J, Wang Y, Tan T, Jain AK (2004) Live face detection based on the analysis of Fourier spectra. In: Biometric technology for human identification, vol 5404. International Society for Optics and Photonics, pp 296–304
38. Määttä J, Hadid A, Pietikäinen M (2011) Face spoofing detection from single images using micro-texture analysis. In: 2011 International joint conference on biometrics (IJCB). IEEE, pp 1–7
39. Matsumoto T (1991) U.S. Patent No. 5,043,922. U.S. Patent and Trademark Office, Washington, DC
40. Pan G, Sun L, Wu Z, Lao S (2007) Eyeblink-based anti-spoofing in face recognition from a generic webcamera
41. Patel K, Han H, Jain AK (2016) Secure face unlock: spoof detection on smartphones. IEEE Trans Inf Forensics Secur 11(10):2268–2283
42. Patel K, Han H, Jain AK (2016) Cross-database face antispoofing with robust feature representation. In: Chinese conference on biometric recognition. Springer, Cham, pp 611–619

43. Patel K, Han H, Jain AK, Ott G (2015). Live face video versus spoof face video: use of moiré patterns to detect replay video attacks. In: 2015 International conference on biometrics (ICB). IEEE, pp 98–105
44. Peixoto B, Michelassi C, Rocha A (2011) Face liveness detection under bad illumination conditions. In: 2011 18th IEEE international conference on image processing (ICIP). IEEE, pp 3557–3560
45. Pinto A, Pedrini H, Schwartz WR, Rocha A (2015) Face spoofing detection through visual codebooks of spectral temporal cubes. IEEE Trans Image Process 24(12):4726–4740
46. Roth J, Tong Y, Liu X (2017) Adaptive 3D face reconstruction from unconstrained photo collections. IEEE Trans Pattern Anal Mach Intell 39(11):2127–2141
47. Roth J, Tong Y, Liu X (2015) Unconstrained 3D face reconstruction. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2606–2615
48. Shang W, Sohn K, Almeida D, Lee H (2016) Understanding and improving convolutional neural networks via concatenated rectified linear units. In: International conference on machine learning, pp 2217–2225
49. Siddiqui TA, Bharadwaj S, Dhamecha TI, Agarwal A, Vatsa M, Singh R, Ratha N (2016) Face anti-spoofing with multifeature videolet aggregation. In: 2016 23rd international conference on pattern recognition (ICPR). IEEE, pp 1035–1040
50. Silberman N, Hoiem D, Kohli P, Fergus R (2012) Indoor segmentation and support inference from RGBD images. In: European conference on computer vision. Springer, Berlin, Heidelberg, pp 746–760
51. Srinivas S, Babu RV (2015) Data-free parameter pruning for deep neural networks. arXiv:1507.06149
52. Sun L, Pan G, Wu Z, Lao S (2007) Blinking-based live face detection using conditional random fields. In: International conference on biometrics. Springer, Berlin, Heidelberg pp 252–260
53. Tan X, Li Y, Liu J, Jiang L (2010) Face liveness detection from a single image with sparse low rank bilinear discriminative model. In: European conference on computer vision. Springer, Berlin, Heidelberg pp 504–517
54. Tran L, Liu X (2018) Nonlinear 3D Face Morphable Model. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7346–7355
55. Tran L, Yin X, Liu X (2017) Disentangled representation learning GAN for pose-invariant face recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1283–1292
56. Wang D, Hoi S, Zhu J (2014) Wlfdb: Weakly labeled face databases, vol 5. Technical report
57. Xu Z, Li S, Deng W (2015). Learning temporal features using LSTM-CNN architecture for face anti-spoofing. In: 2015 3rd IAPR Asian conference on pattern recognition (ACPR). IEEE, pp 141–145
58. Yang J, Lei Z, Liao S, Li SZ (2013) Face liveness detection with component dependent descriptor. ICB 1:2
59. Yang J, Lei Z, Li SZ (2014).Learn convolutional neural network for face anti-spoofing. arXiv:1408.5601
60. Zhang Z, Yan J, Liu S, Lei Z, Yi D, Li SZ (2012). A face antispoofing database with diverse attacks. In: 2012 5th IAPR international conference on biometrics (ICB). IEEE, pp 26–31

Chapter 9
Liveness and Threat Aware Selfie Face Recognition

Geetika Arora, Kamlesh Tiwari and Phalguni Gupta

Abstract Biometric-based human authentication can provide acceptable level of security to mobile devices such as a tablets and smartphones. Face is one of the most popular choices for biometrics on mobile device since the user can conveniently capture his face image by taking a selfie. Like any other security system, selfie face recognition is also vulnerable to attacks wherein an imposter can present photograph of a genuine user to gain an access to the mobile device. Liveness detection is an essential counter-measure to spoof attacks. In adverse scenarios, an attacker can physically force the user to provide his facial image to unlock the phone. In such cases, facial expression detection can act as a counter-measure. This chapter investigates face-based human recognition techniques on mobile devices and highlight methods having liveness and threat awareness.

9.1 Introduction

Mobile devices such as smartphones and tablets have become an essential part of everyone's life due to their usability and versatility. These devices are capable of managing users' schedules and e-commerce. As a result, a lot of important and confidential information remain stored on the device. With the increasing use of mobile devices, access control has become essential to protect from unauthorized access. User authentication enables access control. It can be carried out by using a PIN, passwords, tokens, or patterns. These traditional means of authentication are well

G. Arora (✉) · K. Tiwari
Birla Institute of Technology and Science Pilani, Pilani Campus, Pilani, Jhunjhunu
333031, Rajasthan, India
e-mail: p2016406@pilani.bits-pilani.ac.in

K. Tiwari
e-mail: kamlesh.tiwari@pilani.bits-pilani.ac.in

P. Gupta
National Institute of Technical Teachers' Training and Research (NITTTR) Kolkata,
FC Block Sector 3 Salt Lake, Kolkata 700106, India
e-mail: pg@cse.iitk.ac.in

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision
and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_9

accepted in the society; however, there are some internal and external factors that limit the security while using them [25]. Internal factors refer to user's unsafe behavior. For instance, a user tends to set a simple string as a password to remember; but, this makes it easy to guess. The external factors involve malware or shoulder surfing, in which imposters can acquire the password by their stealthy observation. Biometrics can address these issues based on what you have such as face, and fingerprint, instead of what you do.

Biometric authentication is an automated way of recognizing or verifying a person's identity by using his/her physiological and/or behavioral characteristics [46]. A biometric recognition system is essentially a pattern recognition system that works by obtaining a biometric sample from a subject, extracts the feature set from the sample and matches this feature set with the feature set of the templates stored in the database [10, 41]. Based on the type of application, biometric systems can operate in two modes: verification and identification. In verification mode, a user's acquired biometric sample is matched with his claimed identity, i.e., one-to-one matching. In identification, the acquired sample is matched against all the entries in the database, known as one-to-many matching. Biometric authentication is preferred over other traditional methods because it need not be memorized and it is hard to be spoofed by an imposter. Selfie biometric involves the use of an image captured by the user himself from his mobile device for the purpose of authentication [30, 42].

A face recognition system can be attacked at various stages. Sensor level attack is possible by presenting a fake biometric sample to the sensor. A previously submitted biometric data could also be used for authentication. This is known as replay attack [43]. Another kind of attack is at the feature extractor module in which, it is compelled to choose the features provided by the imposter, instead of extracting features from the genuine user. Attacks can also be executed on the communication channel between the matcher and feature extractor. In this type of attack, imposter steals the extracted features from the sample so as to use it later. The matching module could also be attacked by forcing it to generate a high matching score to bypass the authentication process. Another type of attack is on the database in which the imposter can add a new template, remove or modify the existing templates. The attack can also occur between the matcher and database. The most fatal attack consists of overriding the match score generated by the matcher [11]. All these attacks can be classified as a direct and indirect attack as shown in Fig. 9.1. Direct attack occurs at the sensor level or outside the system. It is when a person pretends to be someone else to acquire unauthorized access to the system. It is also known as spoofing. On the other hand, indirect attacks occur inside the system by manipulating the templates in the database or evading the feature extractor or matcher [31].

A face recognition system has two stages, namely face representation and face matching. During face representation, facial landmarks are extracted by using geometrical descriptors while face matching employs multi-class classifiers [9, 24, 44] to obtain a match between two faces. These are robust to varying expressions and occlusion present in the image. Such attacks pose substantial threat for a security because acquiring a facial image or video of a person from the Internet is not a problem these days. Therefore, face liveness detection algorithms have become essential to

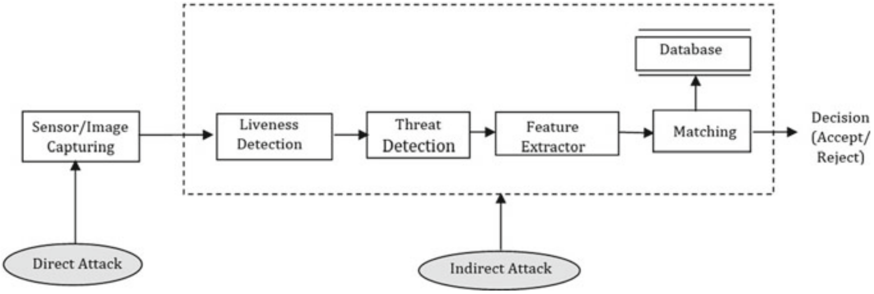


Fig. 9.1 Figure showing points of attacks in face recognition system

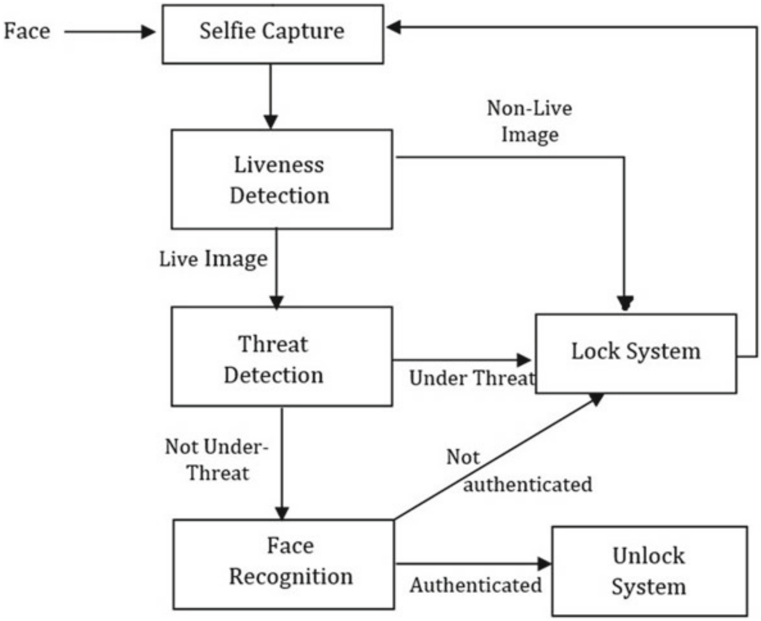


Fig. 9.2 Block diagram: liveness and threat detection in selfie face recognition

detect physical life like signals in the image presented to the system [14]. But only detecting liveness of a face is not sufficient as the genuine person may be threatened to authenticate himself on a device. To address this problem, it becomes necessary to check for under-the-threat authentication after checking for liveness. If the captured facial image is classified as under-threat, then the system remains locked. This process is depicted in Fig. 9.2.

Section 9.2 discusses the important face-based human recognition systems having liveness detection techniques. The threat detection technique has been discussed in Sect. 9.3. Finally, concluding remarks are given in Sect. 9.4.

9.2 Liveness Detection

The selfie face recognition systems are prone to direct attacks. In this type of attack, a fake face image is presented to the system for authentication. To counter this scenario, liveness detection is employed which aim at differentiating a printed photograph or video presented to the face recognition system from a real live image of a person. Liveness detection techniques can be broadly classified into four categories, namely (1) motion-analysis-based, (2) texture-analysis-based, (3) image-quality-analysis-based, and (4) hybrid [32].

9.2.1 Motion-analysis-based Techniques

Motion-analysis-based techniques work by detecting spontaneous movements from the input videos in order to classify the fake and live images. If the input video consists of the expected motion characteristics such as eye blinking, mouth movement, and expression variation, then it is said to correspond to a live user.

An eye-blinking-based liveness detection technique has been proposed in [27]. It models eye-blinking characteristics in an undirected conditional random field (CRF) framework that incorporates different states of eye blinking. Advantage is that it relaxes the conditional independence assumption among the observed states. This helps improve the eye-blink detection, specially in the cases when reflections of light source are present on the glasses. It has been observed that finding an optimal threshold for image binarization is difficult in the case of eye images taken in different resolutions and lighting conditions [13]. To address this limitation, Kim et al. have proposed a technique in [13] to classify closed and open-eye images captured in different conditions by making use of a deep residual convolution neural network. Li et al. have proposed a technique in [15] to explore the artificial intelligence (AI)-based synthesized face videos. These videos are produced by using a series of images generated by a neural network on huge amount of training data [17]. The technique focuses on the detection of lack of eye blinking. It has employed a long-term recurrent convolution neural network (LRCNN) to differentiate between the closed and open eye by considering the previous temporal knowledge. LRCNN have the ability to memorize the previous state due to temporal domain and therefore performs better than the convolution neural network (CNN).

A unidimensional projection of the reduced feature vector technique proposed in [3] applies motion magnification to highlight micro- and macro-facial expressions of the subject. A histogram of the optical flow orientation angle is calculated over local blocks which later are concatenated to obtain a single feature vector, called histogram of oriented optical flows (HOOF). Principal component analysis (PCA) at 95% eigen energy is employed to reduce the feature dimensional. A two-class LDA is used for classification to obtain a unidimensional projection of the reduced feature vector. An approach incorporating dynamic mode decomposition (DMD) has been

Table 9.1 Summary of motion-analysis-based techniques for liveness detection

Author	Strategy	Databases	Parameters and results
Pan et al. [27]	Eye-blinking	Blinking Video Database	Average one-eye rate = 88.8% and average two-eye rate = 95.7% (W = 3)
Bharadwaj et al. [3]	HOOF + LDA	Print Attack Database and Replay Attack Database	HTER = 0 and 1.25% respectively
Kim et al. [13]	Deep residual CNN	NIR eye dataset	EER = 1.88934%
Li et al. [15]	LRCNN	Private (Eye Blinking Video (EBV) dataset)	Area Under the Curve (AUC) = 0.99%
Killioğlu et al. [12]	Pupil tracking	Extended Yale Face Database B	Success ratio = 89.7%
Singh and Arora [36]	Eye-blinking, chin and lip movement	In-house database (65 videos captured at 25 fps for 10 s)	Liveness Detection Rate = 98.98% for front facial images without glasses
Tirunagari et al. [38]	DMD + LBP + SVM	CASIA-FASD	HTER = 21.75%

proposed in [38]. DMD is used to detect motion in the video. LBP pattern is used to detect dynamic patterns and SVM for classification with DMD.

A liveness detection technique based on tracking the direction of the pupil has been proposed in [12]. It has been implemented by observing whether the direction of the pupil matches the LED's position. A liveness detection technique that utilizes multiple liveness detectors has been proposed in [36] uses eye-blink, chin movement, and lip movement for liveness detection. The changes in consecutive frames have been observed to detect any motion. The proposed technique works by comparing pixels of a video frame and static background frame in order to detect the motion.

Motion-analysis-based anti-spoofing techniques have good generalizing ability, but they are computationally heavy and require high user cooperation during the verification process. These techniques fail miserably and report a huge number of authentication failures if user cooperation is not requested. In the case of liveness detectors that rely on spontaneous movements, video display attacks may succeed in fooling the system. A summary of motion-analysis-based techniques with their obtained results is shown in Table 9.1.

9.2.2 *Texture-analysis-based Techniques*

A recaptured image acquired from a 2D photographs of a face may optically look as the images captured from the live face. Therefore, they end up having an overlapping features. To distinguish between the two, suitable feature space needs to be chosen that have sufficient discriminating power to separate the two classes. Texture-analysis-based techniques focus on shape deformation and surface reflection for spoof detection. These artifacts are caused as a result of skin texture difference between spoofed and live face image.

A technique proposed in [21] considers the difference in specular reflections and the surface properties in prints and real images such as pigments. Multi-scale local binary patterns (LBP) have been used to encode micro-texture characteristics difference and spatial information in a feature histogram. The feature histograms are then fed into a support vector machine (SVM) to classify whether the input micro-texture pattern corresponds to a fake or live image. An extension of this technique has been proposed in [22]. It uses two texture features, Gabor wavelets and local binary pattern (LBP) to encode macroscopic information and micro-patterns, respectively, for face description. Further, the low-level information is added to feature description by using a histogram of oriented features (HOG). This feature vector is then transformed into a compact linear form by applying a homogeneous kernel map. This transformed representation is fed into support vector machine that classifies the input into a fake or live image. The previous approach proposed in [22] has been modified by Hassan et al. [8] by using SIFT descriptors along with LBP and Gabor wavelets. The local features descriptors have been extracted using SIFT while the texture features have been extracted using Gabor wavelets and LBP. These three feature vectors are fed into an SVM separately, and the output of three SVM scores are fused to classify if the input is a live image. The results obtained are slightly better than the earlier approaches. An enhanced local binary descriptor (ELBP) proposed in [18] encodes the spatial message and micro-texture difference between the live and fake image. The enhanced feature vector is fed into an SVM for classification. A general technique that works for face, iris, and fingerprint can be found in [1] using locally uniform comparison image descriptor algorithm (LUCID) that analyzes local features. Extracted local patterns are encoded to a feature vector which is used to train SVM for classification between live and fake image. Pinto et al. have proposed a face spoof detection algorithm that takes into account of noise and artifacts generated during the manufacture and recapture of synthetic biometric sample [29]. These artifacts have been characterized by extracting spectral and temporal information from the video. These features are low-level descriptors. Visual code-book is later used to find mid-level descriptors with low-level ones. All the mid-level features are concatenated into one and are fed to SVM for classification.

Many anti-spoof detection algorithms use grayscale images, hence discard the color information that could be important in distinguishing fake images. Keeping this in mind, a facial color texture-based technique has been proposed [5] that takes the chroma component into consideration. Face descriptors have been extracted from three different color bands: RGB, HSV, and YCbCr. Feature descriptors used are: local phase quantization (LPQ), the binarized statistical image features (BSIF), the co-occurrence of adjacent local binary patterns (CoALBP), and the scale-invariant descriptor (SID). These face descriptors extracted from different color channels have been concatenated in order to attain a single enhanced color feature vector. The concatenated feature vectors are passed into an SVM which classifies the input as a

Table 9.2 Summary of texture-analysis-based techniques for liveness detection

Author	Strategy	Databases	Parameters and result
Määttä et al. [21]	Multi-scale Local Binary Pattern	NUAA Photograph Imposter Database	Area Under Curve (AUC) = 0.99%
Määttä et al. [22]	Fusion of LBP, Gabor and HOG	NUAA Photograph Imposter Database	Area Under the Curve (AUC) = 0.999% and Equal Error Rate (EER) = 1.1 %
Hassan et al. [8]	Fusion of SIFT, Gabor and LBP	CASIA and NUAA Imposter dataset	Area Under the Curve (AUC) = 0.9974 and 0.9764% respectively
Liu et al. [18]	Enhanced Local Binary Pattern (ELBP)	NUAA spoofing dataset	Area Under the Curve (AUC) = 0.996% and Accuracy = 95.1%
Boulkenafet et al. [5]	Color Information (Chroma component)	CASIA FASD, Replay-Attack Database, MSU mobile face spoof database	Equal Error Rate (EER) = 2.1, 0.4, 4.9% respectively
Akhtar et al. [1]	LUCID	Print Attack, NUAA, Yale Recaptured, Replay Attack	HTER = 2.880.88%, 1.540.16, 1.900.20 and 5.460.55% respectively
Pinto et al. [29]	Spectral and temporal information and visual code-book	Replay-Attack	HTER = 2.75%

live image or a fake one. A summary of the above-discussed techniques is shown in Table 9.2.

Texture-analysis-based techniques have low computational complexity and therefore have a faster response time. However, they need high-resolution input image to get micro-texture characteristics that becomes a limitation. The generalization ability of these techniques is not good, *i.e.*, when a model is trained on one dataset and tested on another, the performance of spoof detection rapidly degrades [5].

9.2.3 Image-quality-analysis-based Techniques

Techniques lying under this category utilize the fact that there would be a difference in the quality of captured image (*i.e.*, the image acquired from the user) and recaptured images. The recaptured images may have blurriness and would lack in detail and sharpness. These methods assess the quality by using the complete image and hence are more generalized [40, 45]. Galbally et al. [7] have proposed 14 image-quality measures for spoof detection. These measures include signal-to-noise ratio (SNR), normalized absolute error (NAE), structural content (SC), mean angle similarity (MAS), and total edge difference (TED).

An image-distortion-based liveness detection technique has been proposed in [47] by Wen et al. It uses four features, namely specular reflection, chromatic moment, blurriness, and color diversity to construct the feature space. To classify the input image as a live or fake image, an ensemble classifier that consists of multiple SVMs is used. This approach is further extended to multi-frame spoof detection in videos on the basis of majority voting. A liveness detection proposed in [19] selects features based on reflection ratio. It assumes that the reflection ratio of recaptured images is greater than the original image. High-frequency components would be more in genuine face image than recaptured face image and the proportion of image color distribution changes in screen display and print. Three features based on reflection ratio, blurriness, and color proportion are extracted. The feature vectors pertaining to original and recaptured image have been fed into an SVM for classification. A liveness detection method in [28] takes into consideration of image distortion such as surface reflection, shape deformation, color distortion, and moire patterns. It accounts for different image regions, feature descriptors, and color intensity channels and rejects face images on the basis of bezel detection and inter-pupillary distance (IPD) constraint.

In [6], additional source of illumination or a camera has been used to extract a 3D feature vector without explicit reconstruction for liveness detection by analyzing reflection property of the face by modeling it as the Lambertian surface. A Lambertian surface can be defined as the one that reflects light in all possible directions. Therefore, the intensity of the reflected light remains the same even when the camera or the face is moved.

It has been observed that motion analysis techniques involving the use of head rotation and eye blinking are not robust to crude photo-attack in which images of the genuine user are downloaded from the Internet. Therefore, challenge-response authentication can be employed to improve the existing motion-analysis-based techniques. A technique in [26] uses challenge-response authentication along with detecting facial expressions. It uses an Inception-RsNet [37] deep network, namely FaceLiveNet which has two branches corresponding to facial expression recognition and face verification. A deep neural network-based approach proposed in [2], makes use of nonlinear diffusion to differentiate between the original and fake image. The edges obtained from the recaptured image varies from that of the image of the real face. A summary of image-quality-analysis-based techniques for liveness detection is given in Table 9.3.

The image-quality-analysis based techniques have low computational complexity and faster response time. They also have a good generalizing ability but their performance is limited to high-quality input images. With low acquisition quality images, these techniques do not perform well.

Table 9.3 Summary of image-quality-analysis-based techniques for liveness detection

Author	Strategy	Databases	Parameters and result
Wen et al. [47]	Image distortion	Idiap and MSU	True Positive Rate at 0.1% False Acceptance Rate = 92.2 and 94.7%, respectively. In case of training and testing done on same database and = 75.5 and 73.7%, respectively. In case of cross-platform
Luan et al. [19]	Reflection ratio, blurriness and color proportion	NUAA	Accuracy = 98.80%
Patel et al. [28]	Image distortion	Idiap Replay-Attack, CASIA FASD and MSU-MFSD	HTER = 14.6%, EER = 5.88 and 8.41% respectively
Martino et al. [6]	Reflection under different lighting conditions	Private (5010 pairs of stereo images + 7503 pairs of images under different lighting conditions)	Classification Accuracy = 98.9%
Ming et al. [26]	FaceLiveNet	CK+ and OuluCASIA	Accuracy of liveness detection = 100 and 99% respectively
Alotaibi and Mahmood [2]	CNN	NUAA dataset	Accuracy = 98.99%
Galbally and Marcel [7]	14 image quality measures	Replay Attack dataset	HTER = 0.5

9.2.4 *Hybrid Techniques*

Hybrid techniques combine features related to both image-quality and motion analysis for liveness detection. A technique proposed in [45] obtains features related to image quality by analyzing green and red channels of the face image which represents blood flow on the face. The other feature is obtained by approximating the color distribution at local regions of face images, instead of the complete images. The feature space obtained from these two features are then concatenated with a feature obtained from a multi-scale local binary pattern, which is fed into SVM to discriminate between a live and spoofed face images.

Combination of texture and motion-based approach in [3] applies motion magnification to exaggerate micro and macro facial expressions exhibited by a subject. Multi-scale LBP texture features have been used to classify the magnified video of

Table 9.4 Summary of hybrid techniques for liveness detection

Author	Strategy	Databases	Parameters and result
Wang et al. [45]	Texture + motion	NUAA, CASIA, Idiap	Area Under the Curve (AUC) = 99.96, 96.57 and 96.55% respectively
Bharadwaj et al. [3]	Motion + texture	Print Attack Database and Replay Attack Database	HTER = 1.25 and 6.62% respectively
Siddiqui et al. [35]	Motion + texture	CASIA-FASD dataset, MSU mobile face spoofing database and 3D-MAD	EER = 3.14, 0.00 and 0.00% respectively

the subject. Feature space obtained from three LBP configurations ($\mathrm{LBP}^{u2}_{8,1}$, $\mathrm{LBP}^{u2}_{8,2}$ and $\mathrm{LBP}^{u2}_{16,1}$ are concatenated and fed into an SVM with Radial Basis Function (RBF) kernel. An approach that incorporates texture-based features along with motion analysis has been proposed in [35] that cipher video texture and motion. Features of spoof are collected for full frame and segmented face area. Features are then combined for classification. A summary of hybrid techniques with their obtained results is shown in Table 9.4.

9.3 Threat Detection

Only having liveness detection would not be sufficient for security of mobile device. A genuine user may be forcefully asked to undergo the authentication process. If the system could identify a user in threat, it can lock the system and send emergency messages. Threat detection would be based on recognition of facial expression. One can use facial features located at eyelids, eyebrows, lips, cheeks, forehead, and chin to recognize various expressions. Change in facial features has been utilized in [23] to detect expressions. In [23], the facial image after pre-processing is divided into regions for an exclusive localization of feature points. The first half of the region contains eyebrows and eyes. The second half has cheeks, nose, and mouth. The frequency analysis is done on both the parts and a feature vector is obtained. It is passed to a Hopfield neural network for training. This technique has been tested on an in-house dataset and has achieved a success rate of 79.8%. Another technique based on the extraction of features such as lips, eyebrows, or mouth has been proposed in [34]. This has used mouth and eyebrows corners as anchor points for the detection of four facial expressions. It has been tested on JAFFE database [20] and achieved an accuracy of 78% for anger and sad expression and 83% for happy and surprise expression. A feature descriptor, namely local directional pattern number (LDN) has been proposed in [33]. It has been used to encode facial texture information to recognize facial expressions. The face has been divided into several parts, and corresponding LDN feature has been extracted. These different feature vectors are then concatenated to form a global feature vector, which have been used as facial descriptors. An MLP neural network-based facial expression recognition approach has been proposed in [4]. A facial descriptor, namely perceived facial images (PFI) has been used for feature extraction. This technique has been tested on GEMEP FERA, the Cohn-Kanade and the FER-2013 databases.

Under-threat face detection on mobile devices has been addressed by Tiwari et al. [39] along with liveness detection and facial recognition. Threat detection step is performed after the liveness of the input image/video to the face recognition system has been determined. Threat detection involves two parts: (1) recognizing the identity of the user and (2) checking if he/she is under-threat or not. Registration starts with the acquisition of images related to a normal face and an under-threat face of all the users. Two region of interests (ROI) are extracted from the facial image. One containing the forehead and other containing eyes. These regions are then enhanced using gamma correction, a difference of gaussian and contrast equalization. These feature vectors are computed for both the ROIs by using uniform extended local ternary pattern (UELTP) [16]. A global feature vector is obtained by combining the features obtained from both the ROIs. Feature vectors of query sample and the ones stored in the database are compared using Chi-Square distance metric to obtain a matching score. The proposed technique has been tested on two private datasets, namely SmartBioFace and SmartBioThreatFace. The former consists of five frontal face images of a hundred students. The other database contains normal as well as under-threat frontal face images of 100 subjects. Each subject has provided five normal and five under-threat face images in the database. The proposed technique on SmartBioFace database achieved a correct recognition rate (CRR) of 100% and EER of 3.46%. On SmartBioThreatFace dataset, the proposed scheme has obtained an EER of 18.50% along with a CRR of 52.81%.

9.4 Conclusions

Attacks on a facial recognition system have attracted a significant amount of research. Most attacks are direct such as presenting a spoof to the sensor. This chapter reviews spoof detection methods for selfie face recognition. These techniques are classified into four broad categories, namely motion-analysis-based, image-quality-analysis-based, texture-based, and hybrid. It has been observed that motion-based and image-quality-analysis-based methods achieve good generalizing ability. Contextual cues are also useful to detect spoof. For example, a large change in the usual background may be due to attack. Sometimes a finger of the attacker could be seen while he holds the printed photograph. Deep-learning-inspired approaches such as long-term recurrent convolutional neural networks (LRCNN) has been used to detect an eyeblink [15], convolutional neural networks has been used to differentiate between the edges obtained from a fake and a live image [2], FaceLiveNet has been used to detect facial expression/liveness using Inception-RsNet [37]. Deep-learning-based methods have proven to be efficient for classification and generalizing over inter-dataset verification platform. However, they come at an expense of huge training time that may not be suitable for all applications. Only intergrating liveness detection may not be sufficient to ensure mobile device security. As the user could be forced by the attacker to undergo an authentication, therefore threat detection module is also required which typically addresses facial expression classification [20, 33, 34]. The methods achieved good results over the in-house dataset, but a large scale evaluation is needed for statistical significance of the results. Despite a lot of work that has been done toward liveness and threat aware face recognition system, the field is not yet mature. More efficient and accurate techniques are needed as the adaptation of the mobile devices increases.

Appendix

There are certain parameters that are used for evaluating the performance of a face recognition system. Some of the commonly used measures are listed below.

- **One-eye detection rate**: It refers to the rate of correctly detected blinks to the total number of blinks in test data. In this, right and left eyes are calculated separately.
- **Two-eye detection rate**: It is same as one-eye detection rate, but it accounts for the simultaneous blinks of both the eyes for one blink activity.
- **False Acceptance Rate (FAR) and False Rejection Rate (FRR)**: FAR is the likelihood of the system to accept an unauthorized user as an authorized one. FRR, on the other hand, indicates the possibility of the system rejecting an authorized person by considering it as an imposter.
- **Equal Error Rate**: It refers to the value where FAR and FRR are equal and is used to determine their threshold values. The lower the value of EER, higher would be accuracy of a biometric system.
- **Half Total Error Rate (HTER)**: It is computed by averaging the false acceptance rate and false rejection rate.
- **Area Under the Curve (AUC)**: A receiver operating characteristic curve (ROC curve) represents a graphical plot of true positive rate (TPR) against the false positive rate (FPR) at different threshold settings. The area under the ROC curve (AUC) refers to the probability of a randomly chosen positive example being classified as positive with greater suspicion than a randomly selected negative example.

References

1. Akhtar Z, Michelon C, Foresti GL (2014) Liveness detection for biometric authentication in mobile applications. In: 2014 International Carnahan conference on security technology (ICCST), IEEE, pp 1–6
2. Alotaibi A, Mahmood A (2017) Deep face liveness detection based on nonlinear diffusion using convolution neural network. Signal Image Video Process 11(4):713–720
3. Bharadwaj S, Dhamecha TI, Vatsa M, Singh R (2013) Computationally efficient face spoofing detection with motion magnification. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp 105–110
4. Boughrara H, Chtourou M, Amar C-B, Chen L (2016) Facial expression recognition based on a mlp neural network using constructive training algorithm. Multimedia Tools Appl 75(2):709–731
5. Boulkenafet Z, Komulainen J, Hadid A (2016) Face spoofing detection using colour texture analysis. IEEE Trans Inf Forensics Secur 11(8):1818–1830
6. Di Martino JM, Qiu Q, Nagenalli T, Sapiro G (2018) Liveness detection using implicit 3d features. arXiv:1804.06702
7. Galbally J, Marcel S (2014) Face anti-spoofing based on general image quality assessment. In: 2014 22nd International conference on pattern recognition (ICPR). IEEE, pp 1173–1178
8. Hassan MA, Mustafa MN, Wahba A (2017) Automatic liveness detection for facial images. In: 2017 12th International conference on computer engineering and systems (ICCES). IEEE, pp 215–220
9. Huang K-K, Dai D-Q, Ren C-X, Yu-Feng Y, Lai Z-R (2017) Fusing landmark-based features at kernel level for face recognition. Pattern Recogn 63:406–415
10. Jain AK, Ross A, Prabhakar S (2004) An introduction to biometric recognition. IEEE Trans Circ Syst Video Technol 14(1):4–20
11. Jain R, Kant C (2015) Attacks on biometric systems: an overview. Int J Adv Sci Res 1(07):283–288
12. Killioğlu M, Taşkiran M, Kahraman N (2017) Anti-spoofing in face recognition with liveness detection using pupil tracking. In: 2017 IEEE 15th international symposium on applied machine intelligence and informatics (SAMI). IEEE, pp 000087–000092
13. Kim KW, Hong HG, Nam GP, Park KR (2017) A study of deep CNN-based classification of open and closed eyes using a visible light camera sensor. Sensors 17(7):1534
14. Li L, Correia PL, Hadid A (2017) Face recognition under spoofing attacks: countermeasures and research directions. IET Biometrics 7(1):3–14
15. Li Y, Chang M-C, Farid H, Lyu S (2018) In ictu oculi: Exposing ai generated fake face videos by detecting eye blinking. arXiv:1806.02877
16. Liao W-H, Young T-J (2010) Texture classification using uniform extended local ternary patterns. In: 2010 IEEE international symposium on multimedia (ISM). IEEE, pp 191–195
17. Liu M-Y, Breuel T, Kautz J (2017) Unsupervised image-to-image translation networks. Adv Neural Inf Process Syst 700–708
18. Liu X, Lu R, Liu W (2017) Face liveness detection based on enhanced local binary patterns. In: Chinese automation congress (CAC), 2017. IEEE, pp 6301–6305
19. Luan X, Wang H, Ou W, Liu L (2017) Face liveness detection with recaptured feature extraction. In: 2017 International conference on security, pattern analysis, and cybernetics (SPAC). IEEE, pp 429–432
20. Lyons MJ, Budynek J, Akamatsu S (1999) Automatic classification of single facial images. IEEE Trans Pattern Anal Mach Intell 21(12):1357–1362
21. Määttä J, Hadid A, Pietikäinen M (2011) Face spoofing detection from single images using micro-texture analysis. In: 2011 international joint conference on biometrics (IJCB). IEEE, pp 1–7
22. Määttä J, Hadid A, Pietikäinen M (2012) Face spoofing detection from single images using texture and local shape analysis. IET Biometrics 1(1):3–10

23. Manglik PK, Misra U, Maringanti HB et al (2004) Facial expression recognition. In: 2004 IEEE international conference on systems, man and cybernetics, vol 3. IEEE, pp 2220–2224
24. Marcolin F, Vezzetti E (2017) Novel descriptors for geometrical 3d face analysis. Multimedia Tools Appl 76(12):13805–13834
25. Meng W, Wong DS, Furnell S, Zhou J (2015) Surveying the development of biometric user authentication on mobile phones. IEEE Commun Surv Tutorials 17(3):1268–1293
26. Ming Z, Chazalon J, Luoman MM, Visani M, Burie JC Facelivenet: end-to-end networks combining face verification with interactive facial expression-based liveness detection
27. Pan G, Sun L, Wu Z, Lao S (2007) Eyeblink-based anti-spoofing in face recognition from a generic webcamera
28. Patel K, Han H, Jain AK (2016) Secure face unlock: spoof detection on smartphones. IEEE Trans Inf Forensics Secur 11(10):2268–2283
29. Pinto A, Pedrini H, Schwartz WR, Rocha A (2015) Face spoofing detection through visual codebooks of spectral temporal cubes. IEEE Trans Image Process 24(12):4726–4740
30. Poh N, Blanco-Gonzalo R, Wong R, Sanchez-Reillo R (2016) Blind subjects faces database. IET Biometrics 5(1):20–27
31. Ratha NK, Connell JH, Bolle RM (2001) Enhancing security and privacy in biometrics-based authentication systems. IBM Syst J 40(3):614–634
32. Rattani A, Derakhshani R (2018) A survey of mobile face biometrics. Comput Electr Eng 72:39–52
33. Rivera AR, Castillo JR, Chae OO (2013) Local directional number pattern for face analysis: face and expression recognition. IEEE Trans Image Process 22(5):1740–1752
34. Sarode N, Bhatia S (2010) Facial expression recognition. Int J Comput Sci Eng 2(5):1552–1557
35. Siddiqui TA, Bharadwaj S, Dhamecha TI, Agarwal A, Vatsa M, Singh R, Ratha N (2016) Face anti-spoofing with multifeature videolet aggregation. In: 2016 23rd International conference on pattern recognition (ICPR). IEEE, pp 1035–1040
36. Singh M, Arora AS (2018) A novel face liveness detection algorithm with multiple liveness indicators. Wirel Pers Commun 100(4):1677–1687
37. Szegedy C, Ioffe S, Vanhoucke V, Alemi AA (2017) Inception-v4, inception-resnet and the impact of residual connections on learning. In: AAAI, vol 4, p 12
38. Tirunagari S, Poh N, Windridge D, Iorliam A, Suki N, Ho ATS (2015) Detection of face spoofing using visual dynamics. IEEE Trans Inf Forensics Secur 10(4):762–777
39. Tiwari K, Choudhary SK, Gupta P (2016) An efficient face recognition system with liveness and threat detection for smartphones. In: International conference on intelligent computing. Springer, pp 397–406
40. Tiwari K, Gupta P (2014) No-reference fingerprint image quality assessment. In: International conference on intelligent computing. Springer, pp 846–854
41. Tiwari K, Gupta P (2017) Certain challenges in biometrics system development. In: International conference on computational intelligence, cyber security, and computational models. Springer, pp 113–123
42. Trewin S, Swart C, Koved L, Martino J, Singh K, Ben-David S (2012) Biometric authentication on a mobile device: a study of user effort, error and task disruption. In: Proceedings of the 28th annual computer security applications conference. ACM, pp 159–168
43. Uludag U, Jain AK (2004) Attacks on biometric systems: a case study in fingerprints. In: Security, steganography, and watermarking of multimedia contents VI, vol 5306. International Society for Optics and Photonics, pp 622–634
44. Vezzetti E, Marcolin F, Tornincasa S, Ulrich L, Dagnes N (2017) 3d geometry-based automatic landmark localization in presence of facial occlusions. Multimedia Tools Appl 77:1–29
45. Wang S-Y, Yang S-H, Chen Y-P, Huang J-W (2017) Face liveness detection based on skin blood flow analysis. Symmetry 9(12):305
46. Wayman J, Jain A, Maltoni D, Maio D (2005) An introduction to biometric authentication systems. In: Biometric systems. Springer, pp 1–20
47. Wen D, Han H, Jain AK (2015) Face spoof detection with image distortion analysis. IEEE Trans Inf Forensics Secur 10(4):746–761

Part III
Selfie and Soft-Biometrics

Chapter 10
Soft-Biometric Attributes from Selfie Images

Ajita Rattani and Mudit Agrawal

Abstract The aim of this chapter is to discuss the soft-biometric attributes that can be extracted from selfie images acquired from mobile devices. Existing literature suggests that various features in demographics, such as gender and age, in physical, such as periocular and eyebrow, and in material, such as eyeglasses and clothing, have been extracted from selfie images for continuous user authentication and performance enhancement of primary biometric traits. Due to the limited hardware resources, low resolution of front-facing cameras, and the usage of the device in different environmental conditions, factors such as robustness to low-quality data, consent-free acquisition, lower computational complexity, and privacy, favor soft-biometric prediction in mobile devices.

10.1 Introduction to Facial Soft Biometrics

The inception of soft biometrics can be traced back to the mid-nineteenth century when Alphonse Bertillon proposed a very first system based on biometric, morphological, and anthropometric determinations for person identification [1]. Later, Jain redefined soft biometrics as a set of traits providing information about a user, without individually authenticating the subject due to their lack of distinctiveness and permanence [2]. Not necessarily unique to an individual, soft-biometric traits in physical, behavioral, or material form are classifiable in pre-defined human complaint categories [3].

Describable visual attributes are any visual and contextual information that helps represent an image [4]. Typically, visual soft-biometric characteristics can be automatically deducted from primary biometric data, such as predicting gender and age

A. Rattani (✉)
Department of Electrical Engineering and Computer Science,
Wichita State University, Wichita, KS, USA
e-mail: ajita.rattani@wichita.edu

M. Agrawal
Microsoft Corporation, One Microsoft Way, Redmond, WA, USA
e-mail: muditag@microsoft.com

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision
and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_10

from face images and can be broadly classified into demographic, anthropometric, physical, medical, material, and behavioral attributes. Examples of demographic attributes include age, gender, ethnicity, and eye color. Anthropometric and physical traits include body geometry, periocular region, eyebrows, and facial geometry. BMI, wrinkles, health condition, and body weight are an example of medical attributes while eyeglasses, scarf, clothes, lenses, and gait are examples of material and behavioral attributes [2, 3]. There is also an overlap between various attributes. For example, physical attributes like the periocular region and facial geometry could also be indicative of demographics.

Soft biometrics can be used in two ways:

- **As a stand-alone system**: Automated systems based on soft-biometric attributes have drawn significant interest in numerous applications such as surveillance, forensics, human–computer interaction, targeted advertisement systems (e.g., gender-personalized advertising campaign), and search space reduction. Several attributes such as age [5], gender [6], and ethnicity have been proposed for efficient filtering and indexing of biometric databases for increased throughput of the system.
- **In conjunction with primary biometric traits**: Soft biometrics can be used in conjunction with the primary biometric trait for performance enhancement by adding more degrees of freedom to the existing features of the primary biometric trait, especially when primary features are compromised due to image quality. For instance, a hybrid system that combines a face recognition system with soft-biometric attributes such as age, gender, and ethnicity to improve the overall matching accuracy [3]. Woodard et al. [7] have shown that the fusion of periocular and iris biometrics could enhance the performance of the system for non-ideal imagery.

  Soft biometrics can also be employed for continuous or short-term user authentication to ensure that the user primarily authenticated is still the user in control of the device. For instance, Mohammad et al. [8] have used eyebrow region and eyeglasses [9] for short-term user authentication after primary face biometric-based initial authentication offering the best trade-off between computational complexity and accuracy.

Among various visual soft biometrics, selfie soft biometrics is gaining the most popularity due to the recent advancements in front-facing cameras in smartphones. Human faces captured in selfies convey a lot of information in the form of age, ethnicity, jewelry and clothing, emotions, and mental state. This chapter discusses various soft-biometric attributes that can be deduced from selfie images (Fig. 10.1). This chapter is organized into the following sections. The next section focuses on the factors favoring soft-biometric prediction in selfie images. In Sect. 10.3, we discuss the attributes that can be extracted from selfie images, especially those which can be deduced from facial, upper-dorsal and lower-dorsal regions of the user's selfie image, their significance, and the prior work. Finally, we conclude in Sect. 10.4.

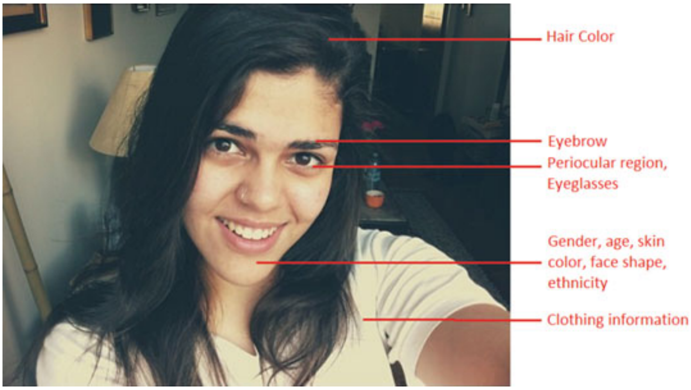

Hair Color
Eyebrow
Periocular region, Eyeglasses
Gender, age, skin color, face shape, ethnicity
Clothing information

Fig. 10.1 Soft-biometric attributes that can be extracted from a selfie face image [10]

10.2 Factors in Favor of Selfie Soft Biometrics

It can be argued that with the latest advancements in machine learning, particularly in deep neural networks, we could extract enough degrees of freedom in primary biometric features like face and iris, to prevent the need for any soft biometrics. However, several factors favor soft-biometric prediction, particularly in the context of mobile devices:

- **Robustness to low-quality data**: Due to the device mobility and operation in an uncontrolled environment (e.g., varying lighting conditions and poses), selfie images captured from mobile devices exhibit substantial degradations resulting in motion blur, poor signal-to-noise ratio, low MTF, and poor uniformity. Figure 10.2 shows sample selfie face images from Adience dataset [11] acquired using mobile devices. Soft-biometric attributes such as gender and age can still be extracted from lower resolution samples captured in an uncontrolled environment when primary biometric trait may not be conclusive for user authentication [12, 13].
- **Consent-free acquisition**: Soft biometrics (such as clothing and eyeglasses) can often be captured without the consent and cooperation of the mobile user. This

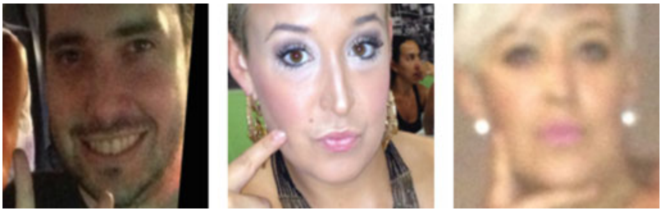


Fig. 10.2 Examples of low-quality face images acquired using mobile devices in an uncontrolled environment

is advantageous for continuous authentication for maintaining the logged-in state after the initial log-in session [14, 15] without asking the user to necessarily cooperate with the device. For example, many face authentication systems require users to look at the camera in a front-parallel position, to best capture facial features. However, once authenticated, the user may not necessarily look at the camera, thereby producing off-axis imagery not suitable for face authentication systems. Soft biometrics could then be used to maintain the logged-in state.

- **Low computational complexity**: Primary biometric authentication is often equipped with presentation attack detection, which needs specialized hardware, for example, FaceID on iPhone uses a patterned light to understand the 3D shape of the face to guard against spoof attacks. Such external light sources, with specialized cameras, increase the power consumption cost, thereby making it difficult to be used in continuous authentication. Soft-biometric attributes processing usually has lower computational complexity than primary biometric data. For instance, processing and feature extraction from eyebrows [8] or periocular region [16] involve a fraction of lower computational complexity than full-face images. In selfie images, using the standard RGB camera instead could lower the cost further down.
- **Privacy**: As soft-biometric characteristics are not unique among subjects, there are lesser privacy and security concerns related to storing soft-biometric attributes in a client–server architecture where the device is used solely for the acquisition of the biometric samples. Feature extraction and matching are performed at the server side [3].

10.3 Soft-Biometric Attributes from Selfie Images

The selfie images can be extracted in three different poses: high, medium, and low as shown in Fig. 10.3. While attributes related to facial features, such as gender, age, eyebrows, and accessories, can be extracted from high and medium poses, the lower pose can obtain rich clothing information from the lower-dorsal region for short-term user authentication [17].

The soft-biometric attributes extracted from selfie images have been categorized in the following subsections.

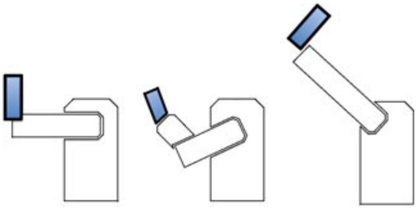


Fig. 10.3 Medium, low, and high poses for selfie image acquisition

10.3.1 Demographic Attributes

10.3.1.1 Age

Age is an important demographic attribute. Humans have an innate ability to reliably estimate the age of their peers based on holistic facial features such as skin texture, wrinkles, skin quality, facial hair, and chin line [3]. Age-based access and targeted advertising are central to various applications. In the context of biometrics, age classification can be employed as a soft-biometric trait in fusion with primary biometric trait to improve the matching accuracy. This is primarily because many biometric algorithms are sensitive to the aging of the subjects. Advances have been made in the form of age-invariant solutions that seek to learn an aging model for age transformation of the input operational image to that of biometric templates [18–20]. These models can either be integrated with existing biometric engines to obtain invariance to the aging effect or could help in age classification from face images by utilizing geometric information, appearance models, and aging pattern subspaces [21]. Despite these studies, the ability to automatically deduce age from a biometric sample is far from an accurate and robust solution. One of the main challenges is that age progression varies among individuals and is influenced by factors such as genetics, health, lifestyle, eating habits, and stress level [21].

In the context of selfie biometrics, a study in [13] proposed convolutional neural networks for age classification from face images acquired using smartphones. The network comprises of only three convolutional layers and two fully connected layers resulting in a small number of neurons. Experimental evidence on full-face images from Adience dataset [11] suggests 1-off the accuracy of 84.7%.

Age classification from ocular images acquired using smartphones was conducted in [5]. This is the first study of its kind as existing studies on age classification have used periocular regions or directly iris images captured in the NIR spectrum instead of mobile RGB captures [22–24]. This is because of the unavailability of the publicly available mobile ocular biometric databases with annotated age information. Authors in [5] used representation learning and proposed a convolutional neural network (CNN) comprising of three convolutional layers and two fully connected layers. The choice of the simpler model was to avoid over-fitting especially for small training datasets. The advantages of such a system include enhanced recognition ability and invariance to aging in smartphone-based ocular recognition (Fig. 10.4).

Further, it provides privacy benefits and reduces computational cost over scanning full-face images for age classification. The accuracy of the proposed CNN model was tested using the cropped version of the recently released Adience dataset [11]. The reported 1-off accuracy of the model was 84.6 ± 1.7%.

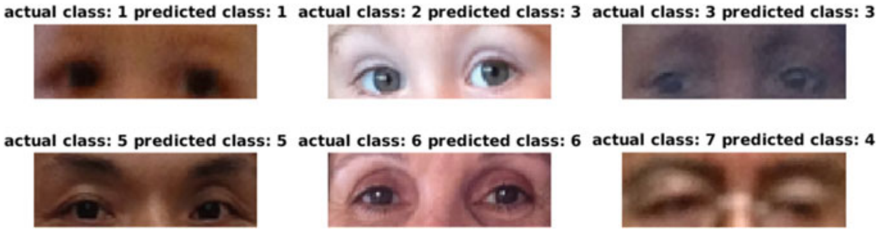

actual class: 1 predicted class: 1
actual class: 2 predicted class: 3
actual class: 3 predicted class: 3
actual class: 5 predicted class: 5
actual class: 6 predicted class: 6
actual class: 7 predicted class: 4

Fig. 10.4 Example of ocular eyebands correctly (in terms of exact and 1-off accuracies) and incorrectly classified into different age groups using the CNN age classification model proposed in [5]

10.3.1.2 Gender

Automated gender estimation has drawn significant interest in numerous applications such as surveillance, human–computer interaction, anonymous customized advertisement systems, image retrieval system as well as in fusion with primary biometric trait to improve the matching accuracy of a biometric system. The number of studies has been proposed for gender estimation from the face, fingerprint, and ocular images. In the context of mobile devices, gender recognition from face [25],[1] keystroke dynamics [26], and accelerometer and gyroscope sensor readings [27] have been studied to enhance the security of the biometric system integrated into mobile devices.

Levi and Hassner [28] proposed convolutional neural networks for gender classification from face images acquired using smartphones. The network uses a simpler model of only three convolutional layers and two fully connected layers with a small number of neurons. Experimental evidence on full-face images from Adience dataset [11] suggests 1-off the accuracy of 86.8%.

Study in [29] proposed gender prediction from ocular images captured in the mobile environment. Authors evaluated local binary patterns (LBP), binary statistical image features (BSIF), local phase quantization (LPQ) and histogram of oriented gradient (HOG) based textural descriptors along with support vector machines (SVM), and multilayer perceptron (MLP) classifiers on a large scale publicly available VISOB dataset captured using mobile devices. The best accuracy of 91.6% was obtained using histogram of gradient (HOG) in combination with MLP classifier.

In another study [6], authors evaluated the use of pre-trained CNNs for gender prediction from ocular images. To this aim, VGGNet, InceptionNet, and ResNet pre-trained on ImageNet dataset were used for feature extraction from selfie ocular images. The extracted features were fed to SVM and MLP-based classifier for training and gender classification on VISOB dataset. The best accuracy of 94.0% was obtained using ResNet for feature extraction in combination with MLP classifier.

[1] https://web.stanford.edu/class/ee368/Project-Spring-1415/Reports/Fu.pdf.

10.3.2 Physical Attributes

10.3.2.1 Eyebrows

Eyebrows are one of the novel biometrics that naturally exists in the human face for all genders. While some studies have shown the potential of eyebrows as standalone biometrics for recognizing individual, others have considered eyebrows as a soft-biometric trait to be used when the primary biometric trait is unavailable due to occlusion (e.g., frames of glasses, glares on glasses, reflective coating on glasses and eyes half-closed) for ocular biometrics [8]. Eyebrow region offers the best trade-off between computational complexity and accuracy in comparison with face or ocular biometrics. This is because the eyebrow region is one-sixth of the full-face region and yet comprises of the feature-rich region on the face.

The advantages of eyebrow-based mobile user authentication include extraction using the front-facing RGB camera available in the mobile device (instead of IR cameras needed for iris recognition), computational efficiency, and fast throughput. Therefore, it can also be employed as a primary or soft-biometric trait in combination with primary biometrics traits such as face and ocular region. Eyebrows can also be used for continuous user authentication to ensure that the user primarily authenticated is still the user in control of the device.

Mohammad et al. [8] developed a method of user authentication using eyebrows for smartphone devices. This is the first large-scale study evaluating the potential of eyebrows biometric for mobile user authentication. The authors used local histogram of oriented gradients (HOG) and global GIST descriptors extracted from eyebrow ROIs with support vector machine (SVM) for user authentication. The experimental results suggest minimum EER of 3.23% on fusing the outputs of left and right eyebrow ROIs for OPPO mobile device in VISOB database. With the continuous advancement in mobile hardware technology, the proposed approach can be used as a viable device-side application for user authentication and as a soft-biometric trait. The performance of eyebrows biometric may be impacted by covariates such as eyeglasses, motion blur, lighting variations, and sometimes by user expressions.

10.3.2.2 Periocular Region

Several studies have been proposed for mobile user recognition based on periocular biometrics which refers to the facial regions in the immediate vicinity of the eye, excluding the pupil, iris, and sclera regions (Fig. 10.5) [30]. Study in [31] used pre-trained VGGNet for feature extraction from the periocular region along with Sparse Augmented Collaborative Representation based Classification. Experimental results on VISOB dataset suggest an accuracy of 99% at the false accept rate of 10^{-3}. In [32], deep feature fusion of iris and periocular recognition was implemented. The proposed method first applies maxout units into the convolutional neural networks (CNNs) to generate a compact representation for each modality and then fuses the discriminative

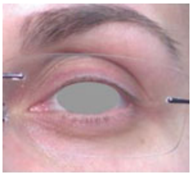


Fig. 10.5 Example of periocular region surrounding the eye

features of two modalities through a weighted concatenation. The parameters of convolutional filters and fusion weights were simultaneously learned to optimize the joint representation of iris and periocular biometrics. Experiments were conducted on CASIA-mobile$V1$ dataset. In [16], Eidinger et al. studied periocular biometrics in a mobile environment. To facilitate this, discrete cosine transform (DCT) based features were extracted from the periocular region and were used to train and test a Gaussian mixture model.

10.3.3 Material Attributes

10.3.3.1 Eyeglasses

Eyeglasses belong to the sub-category of accessories. Patel et al. [14] outlined various approaches that use eyeglasses as a soft-biometric trait for fusion with primary biometric trait and continuous user authentication. A study in [9] proposed two schemes for prescription eyeglasses detection. The first non-learning-based scheme uses Viola–Jones for ocular region of interest (ROI) detection. This is followed by eyeglasses detection, yielding an overall accuracy of 97.9% for VISOB datasets. The second scheme is the learning-based scheme consisting of three main steps (a) ROI detection, (b) fusion of local binary pattern (LBP) and histogram of gradients (HOG) features, and (c) applying classifiers such as support vector machine (SVM), multilayer perceptron (MLP), and linear discriminant analysis (LDA), and eventually combining the output of these classifiers. The best overall accuracy of 100% on VISOB dataset was obtained.

10.3.3.2 Clothing Information

Clothing information has been studied extensively in person re-identification for multi-camera surveillance systems. The advantages of using clothing information for mobile user re-authentication is that clothing ROI is a much larger target compared to the face and eyes, and thus, it can be acquired from the front-facing camera, while a user is naturally interacting with the target application with no explicit cooperation (except an initial consent to allow the method). Further, it is inherently revocable and unlike other soft biometrics, the information stored in the template generally does not compromise user's privacy. Nguyen et al. in [17] investigated the use of clothing information as soft biometrics for short-term mobile user re-authentication. The authors evaluated the feature-level fusion of textural descriptors and deep features extracted using VGGNet in combination with support vector machine (SVM) classifier.

It should be noted that though this method is neither applicable to scenarios where people wear uniform clothing nor when the device camera is not in the general direction of the user's torso. The latter is indeed a benefit since re-authentication should not happen when the user is not naturally interacting with the app that requested the service.

10.3.4 Combination of Demographic, Physical and Material Attributes

Studies in [15, 33] used a combination of gender, age, face shape, hair color, skin color, and the presence of eyeglasses for continuous mobile user authentication using hand-crafted and deep feature representation, respectively. The binary attribute classifiers were trained on each of the soft-biometric attributes using PubFig dataset. The learned classifiers were then applied to the image of the current user of a mobile device to extract the attributes, and then, the authentication was done by directly comparing the difference between the acquired attributes and the enrolled attributes of the user. Experimental investigation on MOBIO dataset suggests EER of 0.19% on the fusion of those aforementioned demographic, physical, and material attributes.

10.3.5 Behavioral Attributes

Apart from demographics, physical and material attributes, behavioral attributes can also be extracted from selfie images. There is growing research which aims at understanding cultural and behavioral patterns using selfie images along with the psychological characteristics of the individuals in selfie images. Musil et al. [34] used the tilt of the head, the side of the face exhibited, mood, and head position to study personality characteristics. The phenomenon of the selfie was explored from three different perspectives:

- **Selfie as a picture**, with the main focus on the visual elements of the picture, their positions, and their relationships. A total of 165 students from a range of fields of study submitted selfies which were coded by three independent raters. The factors considered for rating were background brightness, environmental contexts, tilt of the body, tilt of the head, part of the face, eye contact, frame of the picture, head position in the image, mood, social distance of the subject from the camera, camera

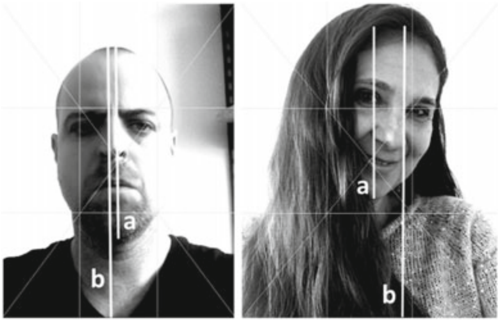


Fig. 10.6 Examples of male and female selfies in Fiji, with horizontal, vertical, and diagonal lines indicated and additional lines for the face-ism calculation (a/b) [34]

position itself, and face-ism defined as the ratio of head length by distance from the top of the head to the lowest part of the body in the photograph.

The modal selfie from the samples was found to be in a vertical frame, taken inside a room (context), with the body and head not tilted, and face centrally exhibited. The head was in the central to the central upper position, and the camera was surprisingly in the left down or left center position with the users posing at a close personal distance with eye contact and positive mood. Apart from these dimensions, male and female selfies differed especially in the categories tilt of the head, tilt of the face, head position, and mood as shown in Fig. 10.6.

- **Selfie as a reflection**, where distinct cues related to the personality characteristics of the individual in the selfie were explored and **selfie as an impression**, where the interpretation of the image in the context of the impression created in others was studied. While no significant relations between the coding cues and psychological constructs (e.g., Big Five personality traits narcissism and femininity–masculinity) were found, entitlement was the only construct that had a significant correlation with the face-ism index.

In the project of Phototrails [35], Hochman and Manovich analyzed 2.3 million Instagram photographs shared by hundreds of thousands of people in 13 global cities to study social, cultural, and political insights about people's activities. In SelfiCity [36] project, smile scores, and extreme poses based on head-tilt angles were compared across various cultures, genders, and ages in five major cities of the world.

It is interesting to note that while there has been a lot of research lately on finding interesting patterns in subgroups using selfie images, and psychological traits in a particular selfie at a specific instance in time, behavioral characteristics of a given individual over time are still an unexplored territory. As online social media continue to capture selfie images of individuals across time, extraction of behavioral attributes of an individual as a soft biometrics remains an interesting direction of research.

10.4 Conclusion

In this chapter, we reviewed existing methods for soft-biometric attribute extraction from selfie images acquired from mobile devices. The methods can be categorized into those extracting demographic, physical, material, and arguably, behavioral attributes from selfie images. We also explored the reasons that favor soft biometrics over primary biometric traits, particularly in the context of mobile devices. These soft-biometric attributes are either used in combination with the primary biometric traits such as the face or ocular region for performance enhancement or continuous user authentication.

As devices around us grow in all forms and factors, biometric authentication is likely to become the norm. This would also bring in multiuser interaction with multiple devices at a given time, instead of a one-to-one interaction which exists today. Soft biometrics would then not only help track authenticated users but would also enhance authentication with various user attributes discussed in this chapter. Depending on a particular scenario and computational cost on a device at a given time, a future system should be able to swiftly switch back and forth between primary biometric authentication (expensive operations which include presentation attack detection) and soft-biometric authentication.

References

1. Rhodes HTF (1956) Alphonse Bertillon: Father of scientific detection. OL 18720791M
2. Jain AK, Dass SC, Nandakumar K (2004) Soft biometric traits for personal recognition systems. In: International conference on biometric authentication, pp 731–738
3. Dantcheva A, Velardo C, Angelo AD, Dugelay JL (2010) Bag of soft biometrics for person identification: new trends and challenges. Multimedia Tools Appl 51(2):739–777
4. Scheirer W, Kumar N, Belhumeur PN, Boult TE (2012) Multi-attribute spaces: calibration for attribute fusion and similarity search. In: The 25th IEEE conference on computer vision and pattern recognition (CVPR), June 2012
5. Rattani A, Reddy N, Derakhshani R (2017) Convolutional neural network for age classification from smart-phone based ocular images. In: IEEE international joint conference on biometrics, Denver, CO, pp 756–761
6. Rattani A, Reddy N, Derakhshani R (2018) Convolutional neural networks for gender prediction from smartphone-based ocular images. IET Biometrics 7:423–430
7. Woodard DL, Pundlik S, Miller PE, Jillela RR, Ross A (2010) On the fusion of periocular and iris biometrics in non-ideal imagery. pp 201–204, Aug 2010
8. Mohammad AS, Rattani A, Derakhshani R (2018) Short-term user authentication using eyebrows biometric for smartphone devices. In: IEEE computer science and electronic engineering conference
9. Mohammad AS, Rattani A, Derahkshani R (2017) Eyeglasses detection based on learning and non-learning based classification schemes. In: IEEE international symposium on technologies for homeland security, pp 1–5
10. Selfiecity database. http://selfiecity.net/selfiexploratory/
11. Hassner T, Harel S, Paz E, Enbar R (2015) Effective face frontalization in unconstrained images. In: IEEE conference on computer vision and pattern recognition, pp 4295–4304

12. Eidinger E, Enbar R, Hassner T (2014) Age and gender estimation of unfiltered faces. IEEE Trans Inf Forensics Secur 9(12):2170–2179 (Special issue on Facial Biometrics in the Wild)
13. Levi G, Hassner T (2015) Age and gender classification using convolutional neural networks. In: IEEE conference on computer vision and pattern recognition (CVPR), Boston
14. Patel VM, Chellappa R, Chandra D, Barbello B (2016) Continuous user authentication on mobile devices: recent progress and remaining challenges. IEEE Signal Process Mag 33:49–61
15. Samangouei P, Patel VM, Chellappa R (2015) Attribute-based continuous user authentication on mobile devices. In: IEEE 7th international conference on biometrics theory, applications and systems. Arlington, VA, pp 1–8
16. de Freitas Pereira T, Marcel S (2015) Periocular biometrics in mobile environment. In: IEEE 7th international conference on biometrics theory, applications and systems (BTAS), pp 1–7, Sept 2015
17. Nguyen H, Sai R, Li Z, Derakhshani R (2018) User re-identification using clothing information for smartphones. In: IEEE international symposium on technologies for homeland security, Woburn, MA, pp 1–6
18. Chen C, Chang Y, Ricanek K, Wang Y (2010) Face age estimation using model selection. In: IEEE computer society conference on computer vision and pattern recognition workshop, pp 93–99
19. Fu Y, Guo G, Huang TS (2010) Age synthesis and estimation via faces. IEEE Trans Pattern Anal Mach Intell 32(11):1955–1976
20. Fu Y, Huang TS (2008) Human age estimation with regression on discriminative aging manifold. IEEE Trans Multimedia 10(4):578–584
21. Dantcheva A, Elia P, Ross A (2015) What else does your biometric data reveal? A survey on soft biometrics. In: IEEE transactions on information forensics and security, pp 1–26
22. Abbasi A, Khan M (2016) Iris-pupil thickness based method for determining age group of a person. Int Arab J Inf Technol 13(6)
23. Erbilek M, Fairhurst M, Abreu MCDC (2013) Age prediction from iris biometrics. In: 5th international conference on imaging for crime detection and prevention. The Institution of Engineering and Technology, Stevenage, pp 1–5
24. Sgroi A, Bowyer KW, Flynn PJ (2013) The prediction of old and young subjects from iris texture. In: IEEE international conference on biometrics, Madrid, pp 1–5
25. Stawska Z, Milczarski P (2016) Gender recognition methods useful in mobile authentication applications. Inf Syst Manage 5(2):248–259
26. Antal M, Nemes G (2016) Gender recognition from mobile biometric data. In: IEEE 11th international symposium on applied computational intelligence and informatics, pp 243–248
27. Jain A, Kanhangad V (2016) Investigating gender recognition in smartphones using accelerometer and gyroscope sensor readings. In: International conference on computational techniques in information and communication technologies, pp 597–602
28. Levi G, Hassner T (2015) Age and gender classification using convolutional neural networks. In: IEEE conference on computer vision and pattern recognition (CVPR) workshops, June 2015
29. Rattani A, Reddy N, Derakhshani R (2017) Gender prediction from mobile ocular images: a feasibility study. In: IEEE international symposium on technologies for homeland security, Waltham, MA, pp 1–6
30. Rattani A, Derakhshani R (2017) Ocular biometrics in the visible spectrum: a survey. Image Vis Comput 59:1–16
31. Alahmadi AA, Aboalsamh HA, Zuair M (2018) Convsrc: Smartphone based periocular recognition using deep convolutional neural network and sparsity augmented collaborative representation. CoRR, abs/1801.05449
32. Zhang Q, Li H, Sun Z, Tan T (2018) Deep feature fusion for iris and periocular biometrics on mobile devices. IEEE Trans Inf Forensics Secur 13(11):2897–2912
33. Samangouei P, Chellappa R (2016) Convolutional neural networks for attribute-based active authentication on mobile devices. In: IEEE 8th international conference on biometrics theory, applications and systems. Niagara Falls, NY, pp 1–8

34. Musil B, Preglej A, Ropert T, Klasinc L, Babic NC (2017) What is seen is who you are: Are cues in selfie pictures related to personality characteristics? In: Frontiers in psychology
35. Hochman N, Manovich L (2013) Zooming into an instagram city: reading the local through social media. First Monday 18(7)
36. Tifentale A, Manovich L (2015) Selfiecity: exploring photography and self-fashioning in social media. Palgrave Macmillan UK, London, pp 109–122

Chapter 11
Sex-classification from Cellphones Periocular Iris Images

Juan Tapia, Claudia Arellano and Ignacio Viedma

Abstract Selfie soft biometrics has great potential for various applications ranging from marketing, security, and online banking. However, it faces many challenges since there is limited control in data acquisition conditions. This chapter presents a super-resolution convolutional neural networks (SRCNNs) approach that increases the resolution of low-quality periocular iris images cropped from selfie images of subject's faces. This work shows that increasing image resolution (2× and 3×) can improve the sex-classification rate when using a random forest classifier. The best sex-classification rate was 90.15% for the right and 87.15% for the left eye. This was achieved when images were upscaled from 150 × 150 to 450 × 450 pixels. These results compare well with the state of the art and show that when improving image resolution with the SRCNN the sex-classification rate increases. Additionally, a novel selfie database captured from 150 subjects with an iPhone X was created (available upon request).

11.1 Introduction

Sex-classification from images has become a hot topic for researchers in recent years since it can be applied to several fields such as security, marketing, demographic studies, among others. The most popular methods for sex-classification are based on face, fingerprint, and iris images. Iris-based sex-classification methods are usually based on near infra-red (NIR) lighting and sensors. This has limited its use since it requires controlled environments and specific sensors. The possibility of using color images to perform iris biometrics has only recently been reported in the literature [1–4].

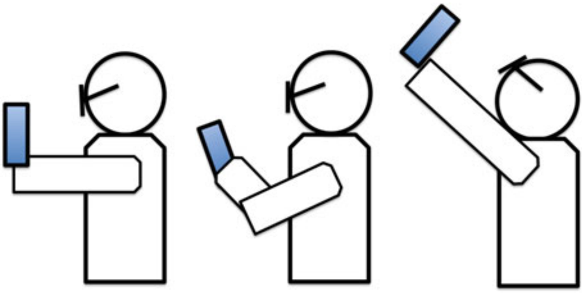
J. Tapia (✉) · C. Arellano
Universidad Tecnologica de Chile—INACAP, Santiago, Chile
e-mail: j_tapiaf@inacap.cl

C. Arellano
e-mail: clarellanov@inacap.cl

I. Viedma
Universidad Andres Bello—DCI, Santiago, Chile
e-mail: i.viedmaescalona@uandresbello.edu

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_11

Fig. 11.1 Representation of different conditions to capture the selfie images. Left: Straight arms. Middle: Half-Straight-arm. Right: Straight arm upper position



These images have traditionally been deemed less suitable for classical iris processing algorithms due to the texture of dark-colored irises not being easily discernible in the visible spectrum.

In order to overcome this limitation, the inclusion of periocular information has been studied and shown to be one of the most distinctive regions of the face. This has allowed it to gain attention as an independent method for sex-classification or as a complement to face and iris modalities under non-ideal conditions. This region can be acquired largely relaxing the acquisition conditions, in contrast to the more carefully controlled conditions usually needed in NIR iris only systems.

Results to date have not just shown the feasibility for sex-classification using VIS periocular iris images but have also reported the feasibility of acquiring other soft biometric information such as; for instance: ethnicity, age, or emotion [5].

In this work, we proposed a method to classify sex from cellphone (selfie) VIS periocular images. This is a challenging task since there is limited control of the quality of the images taken, since selfies can be captured from different distances, light conditions, and resolutions (see Fig. 11.1). Cellphones and mobile devices in general have been widely used for communication, accessing social media, and also for sensitive tasks such as online banking. The use of soft biometrics such as sex-classification in cellphones may be useful for several applications. Real-time electronic marketing, for instance, may benefit from sex-classification by allowing web pages and Apps to offer products according to the person's sex. Data collection tasks may also benefit from discriminating target markets according to sex. Applications in security, on the other hand, may be highly improved by using sex-classification information. It may allow for the protection of users in tasks such as online banking, mobile payment, and sensitive data protection.

Previous work addressing biometric recognition on cellphones includes the use of additional accessories and products specially developed to facilitate this task. An example of such products is Aoptix Stratus,[1] a wrap around sleeve that facilitates NIR iris recognition on the iPhone.

However, these products imply additional cost and only work for specific models of cellphone (iPhone). Therefore, it is important to study a reliable and user-friendly soft biometrics recognition system for all cellphone devices. Furthermore, as biometrics increasingly becomes more widely used, the issue of interoperability is raised

[1] http://www.ngtel-group.com/files/stratusmxds.pdf.

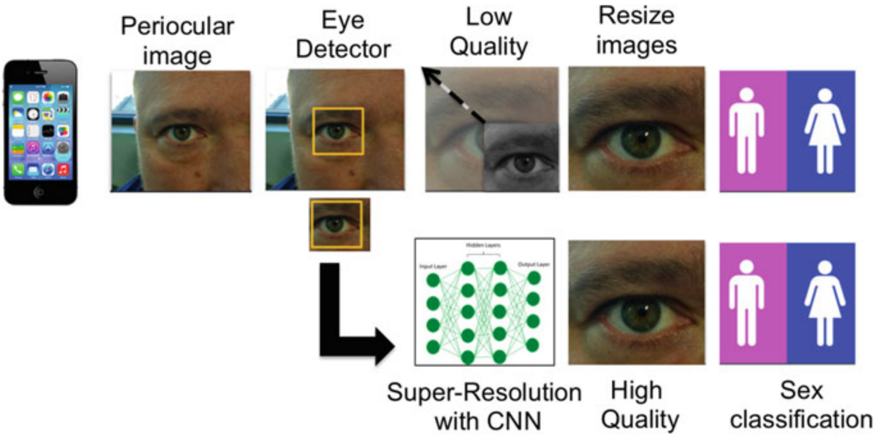


Fig. 11.2 Block diagram of the proposed method. Top: The traditional sex-classification approach. Bottom: The proposed sex classification approach in order to improve the quality of the small images that comes from selfie images

and the exchange of information between devices becomes an important topic of research to validate biometric results, since they should be indifferent to the sensor used to acquire the images [6, 7].

Little work has been reported using periocular VIS cellphone images [8]. They mainly use images that are cropped from selfies. In this context, the resulting periocular iris image has low-quality resolution leading to weak sex-classification results. In this work, a convolutional neural network-super-resolution approach based on [9] was proposed to limit this weakness as it allows the creation of a higher quality version of the same image (see Fig. 11.2). The resulting high-resolution image is then used as input for a random forest algorithm that performs the sex-classification.

This approach is novel as there has not been previous attempts to classify sex from periocular iris cellphone images using super-resolution techniques for increasing the size of the low-quality images that come through selfies.

11.1.1 Sex-Classification from Periocular VIS Images: State of the Art

Sex-classification from periocular VIS images has been reported multiple times in the literature [10–13]. Alonso-Fernandez et al. [2] reviewed the most commonly used techniques for sex-classification using periocular images. They also provided a comprehensive framework covering the most relevant issues in periocular images analysis. They presented algorithms for detecting and segmenting the periocular region, the existing databases, a comparison with face and iris modalities and the identification of the most distinctive regions of the periocular area among other topics. This work gives a comprehensive coverage of the existing literature on soft biometrics analysis from periocular images. A more recent review of periocular iris biometrics from the visible spectrum was made by Rattani et al. [14, 15]. They addressed the subject in terms of computational image enhancement, feature extraction, classification schemes, and designed hardware-based acquisition setups.

Castrillon-Santana et al. [16] also proposed a sex-classification system that works for periocular images. They used a fusion of local descriptors to increase classification performance. They have also shown that the fusion of periocular and facial sex-classification reduces classification error. Experiments were performed on a large face database acquired in the wild where the periocular area was cropped from the face image after normalizing it with respect to scale and rotation.

Kumari et al. [17] presented a novel approach for extracting global features from the periocular region of poor-quality grayscale images. In their approach, global sex features were extracted using independent component analysis and then evaluated using conventional neural network techniques. All the experiments were performed on periocular images cropped from the FERET face database [18].

Tapia et al. [19] trained a small convolutional neural network for both left and right eyes. They studied the effect of merging those models and compared the results against the model obtained by training a CNN over fused left–right eye images. They showed that the network benefits from this model merging approach, becoming more robust toward occlusion and low-resolution degradation. This method outperforms the results obtained when using a single CNN model for the left and right set of images individually.

Previous work addressing sex-classification is summarized in Table 11.1.

Several soft-biometric approaches using periocular iris images captured from mobile devices such as cellphones are presented as follows. Zhang et al. [20] analyzed the quality of iris images on mobile devices. They showed that images are significantly degraded due to hardware limitations and the less-constrained capture environment. The identification rate using traditional algorithms is reduced when using these low-quality images. To enhance the performance of iris identification from mobile devices, they developed a deep feature fusion network that exploits complementary information from the iris and periocular regions. To promote iris recognition research on mobile devices under NIR illumination, they released the CASIA- Iris-Mobile-V1.0 database.

Rattani et al [8] proposed a convolutional neural network (CNN) architecture for the task of age classification. They evaluated the proposed CNN model on the ocular crops of the recent large-scale Adience benchmark for sex and age classification captured using smartphones. The obtained results establish a baseline for deep learning approaches for age classification from ocular images captured by smartphones.

Raghavendra et al. [21] demonstrated a new feature extraction method based on deep sparse filtering to obtain robust features for unconstrained iris images. To evaluate the proposed segmentation and feature extraction method, they employed an iris image database (VSSIRIS). This database was acquired using two different smartphones—iPhone 5 S and Nokia Lumia 1020 under mixed illumination with un-

Table 11.1 Summary of sex-classification methods using images from eyes

Paper	I/P	Source	No. of images	No. of subjects	Type	Acc (%).
Thomas et al. [22]	I	Iris	16,469	N/A	NIR	75.00
Lagree et al. [23]	I	Iris	600	300	NIR	62.17
Bansal et al. [24]	I	Iris	400	200	NIR	83.60
Juan E.Tapia et al. [25]	I	Iris	1500	1500	NIR	91.00
Costa-Abreu et al. [26]	I	Iris	1600	200	NIR	89.74
Tapia et al. [27]	I	Iris	3000	1500	NIR	89.00
Bobeldyk et al. [28]	I / P	Iris	3314	1083	NIR	85.70 (P) 65.70 (I)
Merkow et al. [29]	P	Faces	936	936	VIS	80.00
Chen et al. [30]	P	Faces	2006	1003	NIR/Thermal	93.59
Castrillon et al. [16]	P	Faces	3000	1500	VIS	92.46
Kuehlkamp et al. [31]	I	Iris	3000	1500	NIR	66.00
Rattani et al. [8]	P	Faces	572	200	VIS	91.60
Tapia et al. [13]	I	Iris	10,000 unlabel 3000 labeled	– 1500	NIR	77.79 83.00
Tapia et al. [19]	P	Iris	19,000	1500	NIR	87.26

I Iris images, *P* periocular images, *L* left and *R* right, *Acc* Accuracy

constrained conditions in the visible spectrum. The biometric performance is benchmarked based on the equal error rate (EER) obtained from various state-of-the-art methods and a proposed feature extraction scheme.

11.1.2 Challenges on VIS Cellphone Periocular Images

Selfie biometrics is a new topic only sparsely reported in the literature [15]. Some of the aspects that make sex-classification from selfie images a challenging task are summarized as follows.

Cellphone sensors

The biometrics field is gradually becoming more and more part of daily life thanks to advances in sensor technology for capturing biometric data. More companies are producing and improving sensors for capturing periocular data [32].

Most cameras are designed for RGB and their quality can suffer if they sense light in the IR part of the spectrum. IR blocking filters (commonly known as hot mirrors) are used in addition to Bayer patterns to remove any residual IR. This makes RGB sensors perform poorly when acquiring iris images. Specially when it comes to dark irises.

Due to space, power and heat dissipation limitations, camera sensors on mobile devices are much smaller than traditional iris sensors and the NIR light intensity is much weaker than that of traditional iris imaging devices. Therefore, the image noise on mobile devices is intensive, which reduces the sharpness and contrast of iris texture.

Camera sensor size and focal length are small on mobile devices. As a result images of the iris are often less than 80 pixels in radius, which does not satisfy the requirement described in the international standard ISO/IEC 29794-6.2015 which restricts the iris pupil size to 120 pixels across iris diameters. Moreover, iris radius decreases rapidly as stand-off distance increases. The diameter of the iris decreases from 200 pixels to 135 pixels as the stand-off distance increases by only 10 cm. Although the iris radius in images captured at a distance is usually small, variation with distance is not so apparent because of long focal lengths.

Interoperability across sensors

Several studies have investigated the interoperability of both face and fingerprint sensors. Additionally, there have been reports on sensor safety, illumination, and ease-of-use for iris recognition systems. As of writing, no studies have been conducted to investigate the interoperability of cellphone cameras from various manufacturers using periocular information for sex-classification algorithms. In order to function as a valid sex-classification system, texture sex patterns must prevail independently of the hardware used. The issue of interoperability among cellphones is an important topic in large-scale and long-term applications of iris biometric systems [6, 7, 32].

Non-controlled acquisition environment

In non-constrained image capture settings such as the selfie, it is not always possible to capture iris images with enough quality for reliable recognition under visible light. Periocular iris imaging from cross-sensors allows backward compatibility with existing databases and devices to be maintained while at the same time meet the demand for robust sex-classification capability. The use of the full periocular image helps overcome the limitations of just using iris information, improving classification rates [3, 4, 17].

Periocular cellphone images for biometrics applications are mainly coming from selfie face images. Traditionally, people capture selfie images in multiple places and backgrounds, using selfie sticks, alone or with others. This translates to a high variability of images, in terms of size, light conditions and face pose in the image. To classify sex from a selfie, the periocular iris region from left, right, or both eyes needs to be cropped. Therefore, resulting periocular images usually have very low resolution.

An additional limitation for cellphones is size reduction when images are shared over the Internet. This may affect the accuracy of sex-classification. For example, the iPhone X has a 7 MB selfie frontal camera. But images may be sent over the Internet using four size options: Small (60 Kb), Medium (144 Kb), BIG(684 Kb) and real-size (2 MB).

In this work, a super-resolution CNN algorithm is proposed. This algorithm increases the resolution of images captured using cellphones allowing better sex-classification rates. See Fig. 11.2 (Fig. 11.3).

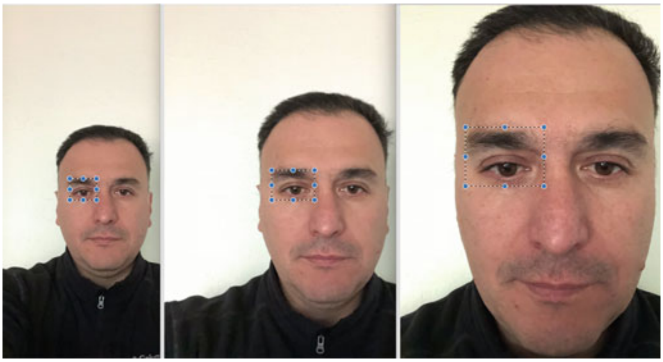


Fig. 11.3 Example of selfie images captured from iPhone X with three different distances. Left: 1.0 mts (Straight arms). Middle: 60 cm (Middle straight arms). Right: 10 cm. (Arms close to the face). Dot squares show the periocular images. All images have the same resolution 2320 × 3088

11.2 Proposed Method for Sex-Classification

In this section, a method for achieving sex-classification from cellphone periocular images is described. The pipeline of this work is shown at the bottom of Fig. 11.2. In Sect. 11.2.1, the data super-resolution convolutional neural network algorithm used for resizing the images in order to increase their resolution is presented. The sex-classifier used afterward is a random forest algorithm which is described in Sect. 11.2.2.

11.2.1 Super-resolution Convolutional Neural Networks

Single-image super-resolution algorithms can be categorized into four types: prediction models, edge-based methods, image statistical methods and patch-based (or example-based) methods. These methods have been thoroughly investigated and evaluated in [33, 34].

In this chapter, a patch-based model to improve resolution of low quality images cropped from selfies is used. The super-resolution using deep learning convolutional neural networks (SRCNNs) algorithm proposed by Dong et al. [9] was implemented. The network directly learns an end-to-end mapping between low- and high-resolution images, with little pre-/post-processing beyond optimization.

The main advantage and most significant attributes of this method are as follows:

1. SRCNNs are fully convolutional, which is not to be confused with fully-connected.
2. An image of any size (provided the width and height will tile) may be input into the algorithm making it very fast in comparison with traditional approaches.

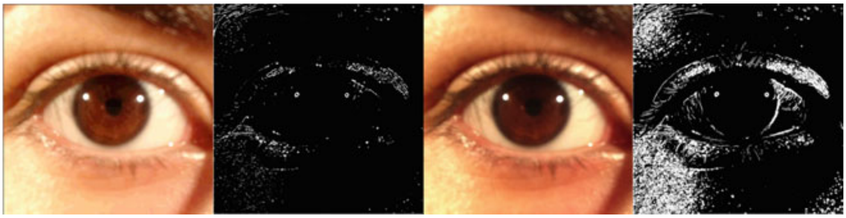


Fig. 11.4 Example of feature maps of CONV1 and CONV2 layers

3. It trains for filters, not for accuracy (see Fig. 11.4).
4. They do not require solving an optimization problem on usage. After the SRCNN algorithm has learned a set of filters, a simple forward pass can be applied to obtain the super resolution output image. A loss function on a per-image basis does not have to be optimized to obtain the output.
5. SRCNNs are entirely an end-to-end algorithm. The output is a higher-resolution version of the input image. There are no intermediate steps. Once training is complete, the algorithm is ready to perform super-resolution on any input image.

The goal while implementing a SRCNNs algorithm is to learn a **set of filters** that allows low-resolution inputs to be mapped to a higher resolution output. Two sets of image patches were created. One of them is a low-resolution patch that is used as the input to the network. And the second one a high-resolution patch that will be the target for the network to predict/reconstruct. The SRCNN algorithm will learn how to reconstruct high-resolution patches from low-resolution input. Figure 11.4 shows filter examples.

11.2.2 Random Forest Classifier

To sex-classify (selfie) periocular images coming from different sensors (cellphones), a random forest classifier (RF) was used. RF algorithm requires a single tuning parameter (Number of trees) making it simpler to use than SVM or neural network algorithms. Furthermore, RF does not require a large amount of data for training like in convolutional neural network algorithm.

RF consists of a number of decision trees. Every node in the decision tree has a condition on a single feature, and it is designed to split the dataset into two. The data with similar response values end up in the same set. The measure for the (locally) optimal condition is called impurity. For classification, the most commonly used impurity measures are the Gini impurity (GDI), the Two Deviance Criterion (TDC) and the Twoing Rule (TR). The Gini's Diversity Index (GDI) can be expressed as follows:

$$Gini\_index = 1 - \sum_{i=1} = p^2(i) \tag{11.1}$$

where, the sum is over the classes i at the node, and $p(i)$ is the observed fraction of classes with class i that reach the node). A node with just one class (a pure node) has Gini index 0; otherwise, the Gini index is positive.

The expression for the deviance of a node using the Two Deviance Criterion (TDC) is defined as follows:

$$TDC\_index = -\sum_{i=1} = p(i)logp(i) \tag{11.2}$$

The TR, on the other hand, can be expressed as:

$$TR\_index = P(L)P(R)(\sum \mid L(i) - R(i) \mid)^2 \tag{11.3}$$

where $P(L)$ and $P(R)$ are the fractions of observations that split to the left and right of the tree, respectively. If the result of the purity expression is large, the split makes each child node purer. Similarly, if the expression is small, the split will make each child node more similar to each other, and hence similar to the parent node. Therefore, in this case, the split does not increase the node purity.

For regression trees, on the other hand, the impurity measure commonly used is the variance. When a tree is trained, the impact of each feature on the impurity of the node can be computed. This allows the features to be ranked according to the impurity measure.

11.3 Experiments and Results

This section describes the experiments performed in order to evaluate sex-classification from periocular VIS images. The databases used for the experiments are first introduced in Sect. 11.3.1. Additionally, a novel hand-made periocular iris image database captured from cellphones (INACAP Database) is presented (available upon request). Pre-processing and data-augmentation steps used for improving performance of the experiments are described in Sect. 11.3.2. Section 11.3.3 describes the process followed to determine the best parameters for the implementation of the SRCNN algorithm. Finally, in Sect. 11.3.4, the experimental setup and results obtained are shown.

11.3.1 Databases

One of the key problems for classifying soft-biometric features such as sex are the small quantity of sex-labeled images. Most databases available were collected for iris recognition applications. They do not, however, usually have soft-biometric information such as sex, age, or ethnicity. In other cases, although this information may have been collected, it is not publicly available since it is considered private information. If only selfie databases were considered, the lack of soft-biometric information is even worse. Most data available on the Internet are unlabeled images. The small amount of sex-labeled selfie images does not allow the training of powerful classifiers such as convolutional neural network and deep learning.

The existing databases used in this work and the novel INACAP database collected for this work are introduced as follows.

11.3.1.1 Existing Databases Used for the Experiments

The following databases were used: CSIP [35], MICHE [36], MODBIO [3]. The **CSIP database** was acquired over cross-sensor setups and varying acquisition scenarios, mimicking the real conditions faced in mobile applications. It considered the heterogeneity of setups that cellphone sensor/lens can deliver (A total of 10 different setups). Four different devices (Sony Ericsson Xperia Arc S, iPhone 4,THL W200 and Huawei Ideos X3 (U8510)) were used and the images were captured at multiple sites. Where artificial, natural, and mixed illumination conditions were used. Some of the images were captured using frontal/rear cameras and LED flash.

The **MICHE Database** captured images using smartphones and tablets such as the iPhone5 (IP5),Galaxy Samsung IV (GS4), and Galaxy Tablet II (GT2).

The **MODBIO** database comprises the biometric data from 152 volunteers. Each person provided samples of face, iris, and voice. There are 16 images for each person. The equipment used for acquisition was a portable handheld device, ASUS transformer Pad TF 300T, with the Android operating system. The device has two cameras—one front and one back. The author used the back camera version TF300T-000128, with 8 MB resolution and autofocus. The sex distribution was 29% females and 71% males. Each image has a size of 640 × 480 pixels.

11.3.1.2 Novel Home-made INACAP-database

This database was collected by students from Universidad Andres Bello (UNAB) and Universidad Tecnologica de Chile - INACAP. This database contains 150 selfie images captured in three different distances according to the position from where the image was taken. We identify three possibles positions and classify the database accordingly:

Set 1: 150 selfies taken while the arm is extended up to front (Fig. 11.1 Left)

Set 2: 150 selfies taken while the arm is bent toward the face (Fig. 11.1 Middle)

Set 3: 150 selfies taken while the arm stretched up from the head (Fig. 11.1 Right)

This is a person disjoint-dataset with 75 female and 75 male selfie images. Table 11.2 shows a summary of the databases used in this chapter.

Table 11.2 VW databases

Dataset	Resolution	No. Images	No. Subjects	F	M	Sensor(s)
CSIP(*) [35]	var. res.	2004	50	9	41	Xperia ArcS, iPhone 4, Th.I W200, Hua U8510
MOBBIO [3]	250×200	800	100	29	71	Cellphones
MICHE [36]	1000×776	3196	92	26	76	iPhone 5
Home-made	2320×3088	450	150	75	75	iPhone X

F represents the number of Female images and *M* the number of Male images; * only left images available

11.3.2 Data Pre-processing and Augmentation

All the images from the databases used present different regions of interest as periocular images. OpenCV 2.10 was used to detect the periocular region and to normalize it by size. An eye detector algorithm was employed to automatically detect and crop the left and right periocular regions. All images were re-sized to 150 × 150 pixels. In those cases where the eye detector failed to select the periocular region, the image was discarded.

To increase the number of images available from the left and the right eye, an image generator function was used. The partition ratio for training, testing, and validation sets was preserved. The dataset was increased from 6000 to 18,000 images for each eye (36,000 images in total) using the following geometric transformations: rotation (in ranges of 10°), width and height shifting (in ranges of 0.2), and zoom range of 15%. All changes were made using the **Nearest fill mode**, meaning the images were taken from the corners to apply the transformation. The mirroring process was not applied since this **may transform** the left eye into a right eye. Care was taken not to mix training and testing examples. See Fig. 11.5.

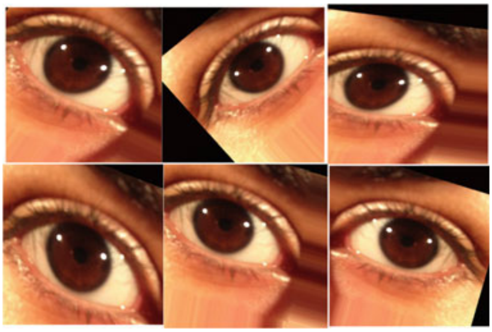


Fig. 11.5 Data-augmentation examples used in order to increase the number of images available to train the classifier

11.3.3 Hyper-parameters Selection

A SRCNNs architecture that consists of only three CONV - RELU layers with no zero-padding was proposed. The first CONV layer learns 64 filters, each of which is 9 × 9. This volume is fed into a second CONV layer where 32 filters of 1 × 1 were used to reduce dimensionality and learn local features. The final CONV layer learns a total of depth channels (which will be 3 for RGB images), each of which are 5 × 5. Finally, in order to measure the error rate, a mean-squared loss (MSE) rather than binary/categorical cross-entropy was used.

The rectifier activation function ReLU controls the nonlinearity of individual neurons and when to activate them. There are several activation functions available. In this work, the suite of activation functions available on the Keras framework was evaluated. However, the best results for these CNNs were achieved where ReLU and Softmax activation functions were used.

In order to find the best implementation for the SRCNNs, the parameters of the CNN such as batch size, epoch, learning rate, among others need to be determined.

Batch size: Convolutional neural networks are in general sensitive to batch size, which is the number of patterns shown to the network before the weights are updated. The batch size has an impact on training time and memory constraint. A set of different batch sizes from $n = 16$ to $n = 512$ in steps of 2^n were evaluated by the SRCNN algorithm.

Epochs: The number of epochs is the number of times that the entire training dataset is shown to the network during training. The number of epochs was tested from 10 up to 100 in steps of 10.

Learning Rate and momentum: The learning rate (LR) controls how much the weights are updated at the end of each batch. The momentum, on the other hand, controls how much the previous update is allowed to influence the current weight update. A small set of standard learning rates from the range $10e - 1$ to $10e - 5$ and momentum values ranging from 0.1 to 0.9 in steps of 0.1 were tried

The selection of the best hyper-parameters of our modified implementation of SR-CNN was found using a grid search fashion. The best classification rate was reached with a batch size of 16, epoch number equal to: 50, LR of $1e - 5$ and momentum equal to 0.9.

According to the size of image used, a stride value equal to 15 was proposed. The patch size used was 25 × 25 pixels. Figure 11.6 shows a example of input and output images from SRCNN.

11.3.4 Experimental Setup and Results

According to the pipeline shown in Fig. 11.2, there are two key processes involved to achieve sex-classification from periocular cellphone images. The super-resolution approach to increase resolution of images and the classifier itself.

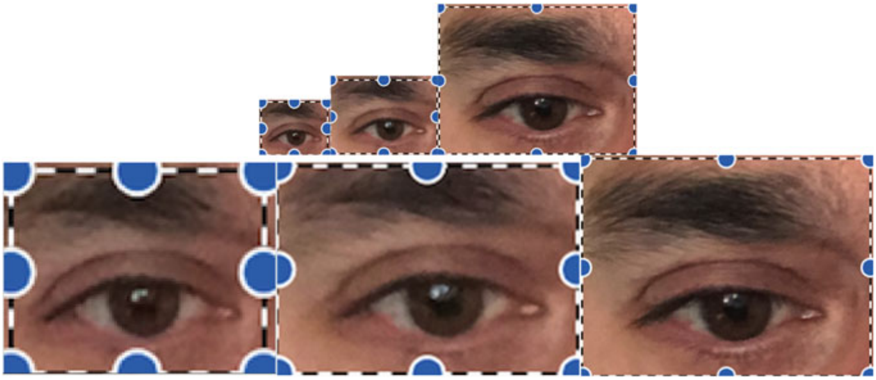


Fig. 11.6 Top: Regular image cropped from face selfie image in three different scales and low-quality images. Botton: Upscaling images generated from SRCNN in high-quality images

For the super-resolution process (SRCNNn), 3000 images taken from existing databases (CSIP, MICHE, MODBIO) were used as input. The algorithm generated 100,000 patches of 25 × 25 pixels. This process allows the filters needed to achieve super-resolution to be estimated. As result, the cropped selfie was transformed from its original dimension of 150 × 150 pixels to high-resolution images of 300 × 300 (2×) and 450 × 450 (3×) pixels (see Fig. 11.6).

The SRCNN algorithm was implemented using Keras and Theano (as the back end), both open-source software libraries for deep learning. The training process was performed on an Intel i7 3.00 GHz processor and Nvidia P800 GPU.

For the sex-classification process, the random forest algorithm was used for all experiments using the three purity measures described in the previous section (Gini (GDI), Two Deviance Criterion (TDC) and the Twoing Rule (TR)). The algorithm was tested using several numbers from the tree (from 100 to 1000). For training, the databases were split into left- and right-eye images. For each eye, the existing databases were used (CSIP, MICHE, MODBIO) with a total of 6000 images plus the augmented data described in Sect. 11.3.2. In total, 18, 000 periocular images for each eye were used (Left and Right). For testing, the INACAP database which contains 450 images was used.

Three experiments were performed to evaluate the sex-classification rate. The first experiment (**Experiment 1**) was used as a baseline for comparison where the inputs are the original 150 × 150 pixel images.

Experiment 2 estimated the sex-classification using the 2X upscaled images from SRCNNs meaning the 300 × 300 pixel images.

Experiment 3 used the 3X upscaled images as input (450 × 450 pixel images).

The rate of sex-classification obtained for all experiments is shown in Table 11.3. Results for the random forest classifier using the three impurity measure and the following number of trees: 100, 300, 500, and 1000 are also reported. The best results for the baseline experiment (Experiment 1) were 68.70% and 71.00% for

Table 11.3 Sex-classification results with random forest classifier using a CSIP, MICHE, and MODBIO dataset for trained and home-made dataset as validation dataset

Model	Tree	Traditional		SRCNN-X2		SRCNN-X3	
		150 × 150		300 × 300		450 × 450	
		Left	Right	Left	Right	Left	Right
		(%)	(%)	(%)	(%)	(%)	(%)
RF-GDI	100	60.65	62.60	69.35	72.15	77.90	78.90
	300	61.35	63.45	68.70	70.25	78.90	78.45
	500	62.45	64.45	70.30	73.40	77.30	79.15
	1000	66.70	68.70	74.45	75.60	79.90	80.25
RF-TR	100	62.25	64.70	75.20	76.15	**83.45**	**83.45**
	300	63.35	66.50	71.20	75.80	**85.15**	**84.30**
	500	64.45	68.70	74.45	76.90	**86.30**	**88.90**
	1000	68.70	71.00	77.50	77.15	**86.70**	**89.45**
RF-TDC	100	63.35	64.50	74.00	74.50	78.90	80.35
	300	64.15	65.90	73.20	75.90	80.15	84.15
	500	64.56	67.70	75.50	76.80	**84.05**	**89.25**
	1000	68.70	70.15	76.20	76.50	**87.15**	**90.15**

SRCNN-X2 represents of result with two times upscaling. SRCNN-X3 represents of result with three times upscaling

the left and right periocular images, respectively. Results improved as the image resolution increased. The best sex-classification rate (90.15% for the right eye and 87.15%) was achieved when 450 × 450 pixel images were used (SRCNN-3) and the RF algorithm was implemented using the TDC metric. These results are competitive with the state of the art and shows that when improving image resolution with SRCNN the sex-classification rate from periocular selfie images also improved.

11.4 Conclusion

Selfie biometrics is a novel research topic that has great potential for multiple applications ranging from marketing, security, and online banking. However, it faces numerous challenges to its use as there is only limited control over data acquisition conditions compared to traditional iris recognition systems, where the subjects are placed in specific poses in relation to the camera in order to capture an effective image. When using selfie images, we do not just deal with images taken from challenging environments, conditions, and settings but also with low resolution since periocular image are mainly cropped from images of the entire face.

This chapter is preliminary work that demonstrates the feasibly of sex-classification from cellphone (Selfie) periocular images. It has been shown that when using super-resolution convolutional neural networks for improving the resolution of periocular images taken from selfies, sex-classification rates can be improved.

In this work, a random forest classifier algorithm was used. However, in order to move forward in this topic, it is necessary to create new sex-labeled databases of periocular selfie images. This would allow the use of better classifiers such as those based on deep learning. An additional contribution of this work, is a novel hand-made database (INACAP) that contains 450 sex-labeled selfie images captured with an iPhone X (Available upon request).

Acknowledgements This research was funded by CONICYT, through grant FONDECYT - 11170189 And Universidad Tecnologica de Chile - INACAP. Further thanks to NVIDIA Research program to donate a P800 GPU card and make possible this research.

References

1. Proenca H, Alexandre LA (2007) The nice.I: Noisy iris challenge evaluation—part I. In: First IEEE international conference on biometrics: theory, applications, and systems (BTAS 2007), pp 1–4, Sept 2007
2. Alonso-Fernandez F, Bigun J (2016) A survey on periocular biometrics research. Patt Recogn Lett 82(2):92–105
3. Sequeira AF, Monteiro JC, Rebelo A, Oliveira HP (2014) MobBIO: a multimodal database captured with a portable handheld device. In: Proceedings of the 9th international conference on computer vision theory and applications (VISIGRAPP 2014), pp 133–139
4. Nigam I, Vatsa M, Singh R (2015) Ocular biometrics. Inf Fusion 26:1–35
5. Dantcheva A, Elia P, Ross A (2015) What else does your biometric data reveal? A survey on soft biometrics. IEEE Trans on Inform Forens Secur. ISSN: 1556-6013
6. Boyce C, Ross A, Monaco M, Hornak L, Li Z (2006) Multispectral iris analysis: A preliminary study. In: Conference on computer vision and pattern recognition workshop (CVPRW ’06), pp 51–51, June 2006
7. Pillai J, Puertas M, Chellappa R (2014) Cross-sensor iris recognition through kernel learning. IEEE Trans Patt Anal Mach Intell 36:73–85
8. Rattani A, Reddy N, Derakhshani R (2017) Convolutional neural network for age classification from smart-phone based ocular images. In: 2017 IEEE international joint conference on biometrics (IJCB 2017), Denver, CO, USA, pp 756–761, 1–4 Oct 2017
9. Dong C, Loy CC, He K, Tang X (2016) Image super-resolution using deep convolutional networks. IEEE Trans Patt Anal Mach Intell 38:295–307
10. Ahuja K, Islam R, Barbhuiya FA, Dey K (2016) A preliminary study of cnns for iris and periocular verification in the visible spectrum. In: 23rd international conference on pattern recognition (ICPR), pp 181–186, Dec 2016
11. Tapia J, Viedma I (2017) Gender classification from multispectral periocular images. In: 2017 IEEE international joint conference on biometrics (IJCB), pp 805–812, Oct 2017
12. Tapia J, Aravena C (2017) Gender classification from nir iris images using deep learning. In: Bhanu B, Kumar A (Eds) Deep learning for biometrics. Springer International Publishing, Berlin, pp 219–239
13. Tapia J (2017) Gender classification from near infrared iris images
14. Rattani A, Derakhshani R (2017) Ocular biometrics in the visible spectrum: A survey. Image Vis Comput 59:1–16
15. Rattani A, Derakhshani R (2017) On fine-tuning convolutional neural networks for smartphone based ocular recognition. In: 2017 IEEE international joint conference on biometrics (IJCB 2017), Denver, CO, USA, pp 762–767, 1–4 Oct 2017

16. Castrillon-Santana M, Lorenzo-Navarro J, Ramon-Balmaseda E (2016) On using periocular biometric for gender classification in the wild: an insight on eye biometrics. Patt Recogn Lett 82(2):181–189
17. Kumari S, Bakshi S, Majhi B (2012) Periocular gender classification using global ICA features for poor quality images. Proc Eng 38:945–951
18. Wechsler PH, Huang J, Rauss PJ (1998) The feret database and evaluation procedure for face-recognition algorithms. Image and Vis Comput 16:295–306
19. Tapia J Aravena CC (2018) Gender classification from periocular nir images using fusion of cnns models. In: 2018 IEEE 4th international conference on identity, security, and behavior analysis (ISBA), pp 1–6, Jan 2018
20. Zhang Q, Li H, Sun Z, Tan T (2018) Deep feature fusion for iris and periocular biometrics on mobile devices. IEEE Trans Inform Forens Secur 13:2897–2912
21. Raja KB, Raghavendra R, Vemuri VK, Busch C (2015) Smartphone based visible iris recognition using deep sparse filtering. Patt Recogn Lett 57:33–42
22. Thomas V, Chawla N, Bowyer K, Flynn P (2007) Learning to predict gender from iris images. In: First IEEE international conference on biometrics: theory, applications, and systems (BTAS 2007), pp 1–5
23. Lagree S, Bowyer K (2011) Predicting ethnicity and gender from iris texture. In: IEEE international conference on technologies for homeland security (HST), pp 440–445, Nov 2011
24. Bansal A, Agarwal R, Sharma RK (2012) SVM based gender classification using iris images. In: Fourth international conference on computational intelligence and communication networks (CICN), pp 425–429, Nov 2012
25. Tapia JE, Perez CA, Bowyer KW (2014) Gender classification from iris images using fusion of uniform local binary patterns. In: European conference on computer vision (ECCV). Soft biometrics workshop
26. Costa-Abreu MD, Fairhurst M, Erbilek M (2015) Exploring gender prediction from iris biometrics. In: International conference of the biometrics special interest group (BIOSIG), pp 1–11
27. Tapia J, Perez C, Bowyer K (2016) Gender classification from the same iris code used for recognition. IEEE Trans Inform Forens Secur 99:1–1
28. Bobeldyk D, Ross A (2016) Iris or periocular? exploring sex prediction from near infrared ocular images. Lectures Notes in Informatics (LNI). Bonn, Gesellschaft fur Informatik, p 2016
29. Merkow J, Jou B, Savvides M (2010) An exploration of gender identification using only the periocular region. In: Fourth IEEE international conference on biometrics: theory applications and systems (BTAS), pp 1–5
30. Chen C, Ross A (2011) Evaluation of gender classification methods on thermal and near-infrared face images. In: International joint conference on biometrics (IJCB), pp 1–8. IEEE
31. Kuehlkamp A, Becker B, Bowyer K (2017) Gender-from-iris or gender-from-mascara? In: 2017 IEEE Winter conference on applications of computer vision (WACV), pp 1151–1159, Mar 2017
32. Connaughton R, Sgroi A, Bowyer K, Flynn P (2012) A multialgorithm analysis of three iris biometric sensors. IEEE Trans Inform Forens Secur 7:919–931
33. Glasner D, Bagon S, Irani M (2009) Super-resolution from a single image. In: 2009 IEEE 12th international conference on computer vision, pp. 349–356, Sept 2009
34. Wei Z, Xiaofeng B, Fang H, Jun W, Mongi AA (2018) Fast image super-resolution algorithm based on multi-resolution dictionary learning and sparse representation. J Syst Eng Electron 29:471–482
35. Santos G, Grancho E, Bernardo MV, Fiadeiro PT (2015) Fusing iris and periocular information for cross-sensor recognition. Pattern Recogn Lett 57:52–59
36. De Marsico M, Nappi M, Riccio D, Wechsler H (2015) Mobile iris challenge evaluation (miche)-i, biometric iris dataset and protocols. Pattern Recogn Lett 57:17–23

Chapter 12
Active Authentication on Mobile Devices

Pramuditha Perera and Vishal M. Patel

Abstract In recent years, we have witnessed a significant growth in the use of mobile devices such as smartphones and tablets. In this context, security and privacy in mobile devices becomes vital as the loss of a mobile device could compromise personal information of the user. To deal with this problem, Active Authentication (AA) systems have been proposed in the literature where users are continuously monitored after the initial access to the mobile device. In this chapter, we provide a survey of recent face-based AA methods.

12.1 Introduction

Traditional methods for authenticating users on mobile devices are based on explicit authentication mechanisms such as passwords/ pin numbers or secret patterns. Studies have shown that users often choose a simple, easily guessable password like "12345," "abc1234," or even "password" to protect their data [1, 2]. As a result, hackers could easily break into many accounts just by trying most commonly used passwords. On the other hand, when a secret pattern is used to gain initial access to the mobile device, the user would draw the same pattern multiple times on the screen over the time. It has been shown that with special lighting and high-resolution photograph, one can easily deduce the secret pattern (see Fig. 12.1) [3] using the oily residues or smudges left on the screen.

Furthermore, recent studies have shown that about 34% or more users did not use any form authentication mechanism on their devices [4–7]. In these studies, inconvenience was cited to be one of the main reasons why users did not use any authentication mechanism on their devices [6, 7]. Moreover, [7] demonstrated that mobile device users considered unlock screens unnecessary in 24% of the situations and they spent up to 9% of time they use their smartphone to deal with unlock screens.

P. Perera · V. M. Patel (✉)
Johns Hopkins University, Baltimore, MD, USA
e-mail: vpatel36@jhu.edu

P. Perera
e-mail: pperera3@jhu.edu

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision
and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_12

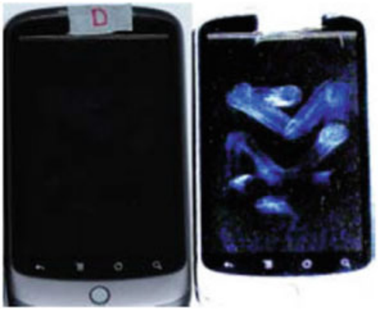


Fig. 12.1 Smudge attack [3]. Secret pattern can be determined with special lighting and high-resolution camera

Furthermore, as long as the mobile phone remains active, typical devices incorporate no mechanisms to verify whether the user originally authenticated is still the user in control of the device. Thus, unauthorized individuals could potentially obtain access to personal information of the user if a password is compromised or if the user does not exercise adequate vigilance after initial authentication on a device.

In order to overcome these issues, both biometrics and security research communities have developed techniques for continuous authentication on mobile devices. These methods essentially make use of the physiological and behavioral biometrics using the built-in sensors and accessories such as gyroscope, touchscreen, accelerometer, orientation sensor, and pressure sensor to continuously monitor the user identity. For instance, physiological biometrics such as face can be captured using the front-facing camera of a mobile device and can be used to continuously authenticate a mobile device user. On the other hand, sensors such as gyroscope, touchscreen, and accelerometer can be used to measure behavioral biometric traits such as gait, touch gestures, and hand movement transparently. Figure 12.2 highlights some of the sensors and accessories available in a modern mobile device. These sensors are capable of providing raw data with high precision and accuracy. Therefore, they can be used to monitor three-dimensional device movement, device positioning, and changes in ambient environment near the device. Note that the terms continuous authentication, Active Authentication [8], implicit authentication [9, 10], and transparent authentication [11] have been used interchangeably in the literature.

12.2 Common AA Approaches

Figure 12.3 shows the typical setup of a biometrics-based mobile device continuous authentication system [12]. Biometric modalities such as gait, face, keystroke, or voice are measured by the sensors and accessories that are available in the mobile device. Then, the biometric system determines whether these biometric traits correspond to a legitimate user or not. If the features do correspond to the legitimate user, the biometric system will continue to process new incoming data. However, if the biometric system produces a negative response then the system will prompt the

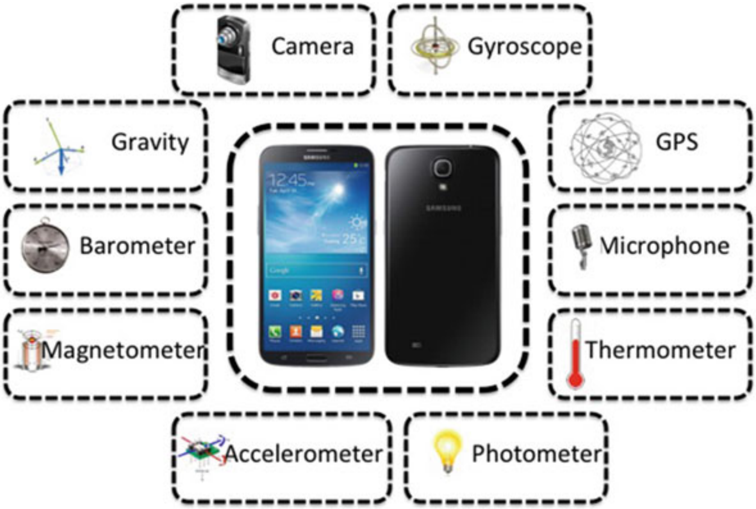


Fig. 12.2 Sensors and accessories available in a mobile device. Raw information collected by these sensors can be used to continuously authenticate a mobile device user

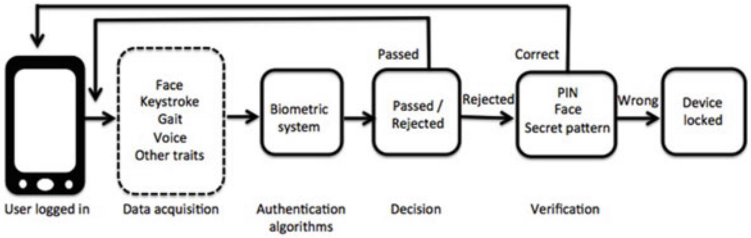


Fig. 12.3 A biometrics-based mobile continuous authentication framework [12]

user to verify his or her identity by using a traditional explicit authentication method. If the user is able to verify his identity, then he will be allowed to use the mobile device. Otherwise, the device will be locked. In a practical continuous authentication system, this entire process happens in real time.

A plethora of mobile continuous authentication methods have been proposed in the literature. Screen-touch gestures are one of the earliest modalities proposed for Active Authentication. Screen-touch gestures are basically the way users swipe their fingers on the screen of mobile devices. They have been used to continuously authenticate users while users perform basic operations on the phone [13–18]. In these methods, a behavioral feature vector is extracted from the recorded screen-touch data and a discriminative classifier is trained on these features for authentication. Touch gestures along with the micro-movement of the device caused by user's screen-touch actions have also been used for authentication in [19]. Stylometry, GPS location, Web browsing behavior, and application usage patterns were used in [20] for continuous authentication. Face-based continuous user authentication has also been proposed in [21–24]. Gait as well as device movement patterns measured by the smartphone accelerometer were used in [25, 26] for continuous authentication. Fusion of speech and face was proposed in [21] while [27] proposed to fuse face images with the inertial measurement unit data to continuously authenticate the users. A low-rank representation-based method was proposed in [28] for fusing touch gestures with faces for continuous authentication. A domain adaptation method was proposed in [29] for dealing with data mismatch problem in continuous authentication. Some of the other continuous authentication methods are based on Web browsing behavior [30], behavior profiling [31], text-based [32, 33], and body prints [34].

12.3 Face-Based AA Methods

The face modality is one of the widely used biometric modalities in Active Authentication. Such systems typically consist of three main stages. In the first stage, faces are detected from the images or videos captured by the front-facing cameras of smartphones. Then, holistic or local features are extracted from the detected faces. Finally, these features are passed on to a classifier for authentication. A number of different methods have been proposed in the literature for detecting and recognizing faces on mobile devices. In what follows, we provide a brief overview of recent face-based AA methods that have been proposed recently in the literature [23, 35–41].

In [24], the feasibility of face and eye detection on mobile phones was evaluated using AdaBoost cascade classifiers with Haar-like and LBP features as well as a skin color-based detector. On a Nokia N90 mobile phone that has an ARM9 220 MHz processor and a built-in memory of 31 MB, their work reported that the Haar + AdaBoost method can detect faces in 0.5 s from 320 × 240 images. This approach, however, is not effective when wide variations in pose and illumination are present or the images contain partial or clipped images. To deal with these issues, a deep convolutional neural network (DCNN)-based method was recently developed in [42] for detecting faces on mobile platforms. In this method, deep features are first extracted using the first five layers of AlexNet [43]. Different-sized sliding windows are considered, to account for faces of different sizes, and an SVM is trained for each window size to detect faces of that particular size. Then, detections from all the SVMs are pooled together and some candidates are suppressed based on overlap criteria. Finally, a single bounding box is generated as the output by the detector. It was shown that this detector is quite robust to illumination change and is able to detect partial or extremely profile faces. A few sample positive detections from the UMDAA dataset [22] are shown in Fig. 12.4. The DCNN-based detections are marked in red, while the ground truth is in shown yellow. Another part-based method for detecting partial and occluded faces on mobile devices was developed in [44]. This method is based on detecting facial segments in the given frame and clustering them to obtain the region that is most likely to be a face.

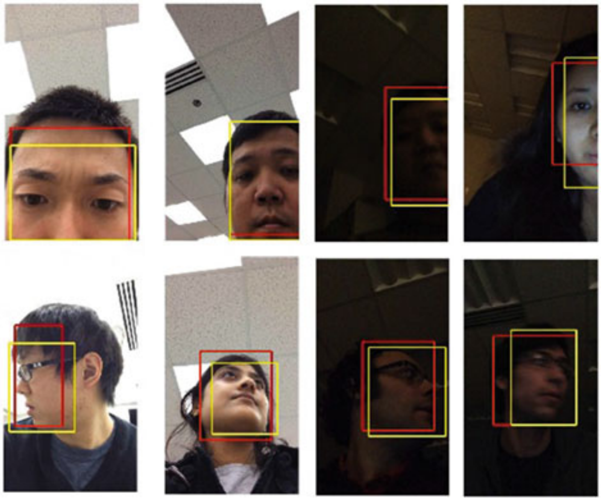


Fig. 12.4 Examples of positive detections with pose variations and occlusion on the UMDAA dataset. The detector's output is in red, while ground truth is in yellow [42]

Several works in the literature have used face recognition-based algorithms to perform Active Authentication. In [45], a AA method was proposed based on one-class SVM. In their approach, faces are detected using the Viola–Jones detector [46]. Histogram equalization is then applied on the detected images to normalize the effect of illumination. Finally, two-dimensional Fourier transform features are extracted from the normalized images and fed into one-class SVM for authentication. In [24], a face and eye detection scheme has been introduced along with a LBP feature-based face recognition method designed for mobile devices. It was shown that their proposed continuous face authentication system can process about 2 frames per second on a Nokia N90 mobile phone with an ARM9 processor with 220 MHz. Average authentication rates of 82 and 96% for images of size 40 × 40 and 80 × 80, respectively, were reported in [24]. In [22], a number of different face recognition methods were evaluated on a dataset of 750 videos from 50 users collected over three sessions with different illumination conditions.

12.3.1 Attribute-Based AA

Visual attributes are essentially labels that can be given to an image to describe its appearance [47]. A facial attribute-based continuous authentication method was recently proposed in [23, 38]. Figure 12.5 gives an overview of this method. Given a face image sensed by the front-facing camera, pre-trained attribute classifiers are used to extract a 44-dimensional attribute feature. The binary attribute classifiers

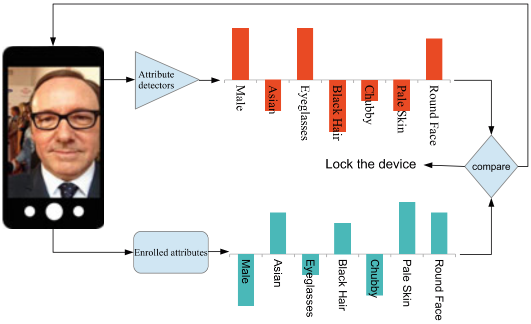


Fig. 12.5 Overview of the attribute-based authentication method proposed in [23]

are trained using the PubFig dataset [47] and provide compact visual descriptions of faces. The score is determined by comparing extracted attribute features with the features corresponding to the enrolled user. These score values are essentially used to continuously authenticate a mobile device user. Furthermore, it was shown that the attribute-based method can be fused LBP features [24] to obtain an improved performance.

This method was later extended in [39] where DCNNs were used to predict attributes for AA. In particular, a multi-task, part-based DCNN architecture was proposed for attribute detection. It was shown in [39] that this method can outperform the previously presented attribute-based methods as well as baseline LBP method for face-based mobile AA. Furthermore, effectiveness of this architecture was also demonstrated in terms of speed and power consumption by deploying it on an actual mobile device.

12.3.2 Extreme Value Analysis for Mobile AA

In principle, the primary goal of an authentication system is to ensure information security through intruder prevention. In order to prevent intrusions, an authentication mechanism should operate with a very low degree of false alarms. In [35], a special emphasis was given to the AA systems at the low false alarm region (from 0.001 to 0.1) and a new performance enhancement mechanism in this region for unimodal mobile AA systems was presented based on the statistical Extreme Value Theory (EVT).

Figure 12.6 gives an overview of this EVT-based AA system. A typical AA system extracts features of a probe and compares them against the enrolled features. In this system, distribution of the probability scores is also obtained in the enrollment phase. Tail of the probability score distribution is modeled using the EVT and is used together with the similarity score generated in the standard AA system to enhance the performance of the standard system. It is interesting to note that this EVT-based mechanism is independent of sensors and features used in the

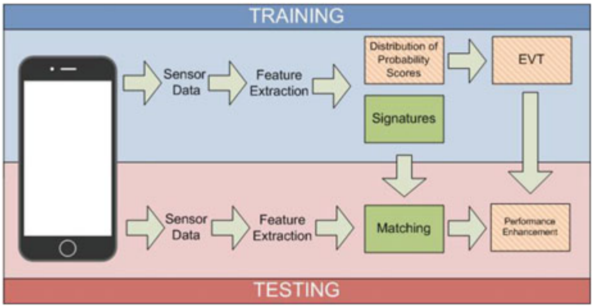


Fig. 12.6 Overview of the EVT-based AA system. Non-shaded blocks represent typical components of an AA system. Shaded components are the additions for performance enhancement [35]

underline AA system. Therefore, any existing AA system can be extended by incorporating this performance enhancement scheme. Experiments were conducted on three publicly available AA datasets, and it was shown that the new method can improve performance of the existing face and touch gesture-based AA systems.

12.3.3 One-Class Classification

Due to unavailability of training samples from negative classes, AA can be viewed as an one-class classification problem. To this end, a Single-class Minimax Probability Machine (1-MPM)-based solution called Dual Minimax Probability Machines (DMPM) for AA applications was recently introduced in [48]. In contrast to 1-MPM, this method has two notable differences.

(1) An additional hyper-plane is learned to separate training data from the origin by taking into account maximum data covariance.
(2) The possibility of modeling the underline distribution of training data is considered as a collection of sub-distributions.

Intersection of negative half-spaces defined by the two learned hyper-planes is considered to be the negative space during inference. The effectiveness of this mechanism was demonstrated by performing evaluations on three publicly available face-based AA datasets. In particular, it was shown that the decision boundary found by this method was indeed better than the decision boundary produced by 1-MPM. In all datasets, DMPM method demonstrated an improvement of 4–6% compared to 1-MPM.

In another work [49], a new DCNN-based one-class classification algorithm was recently introduced and was evaluated on AA application. Figure 12.7 gives an overview of the proposed CNN-based approach for one-class classification. The overall network consists of a feature extractor network and a classifier network. The feature extractor network essentially embeds the input target class images into a feature space. The extracted features are then appended with the pseudo-negative class data and generated from a zero-centered Gaussian in the feature space. The appended

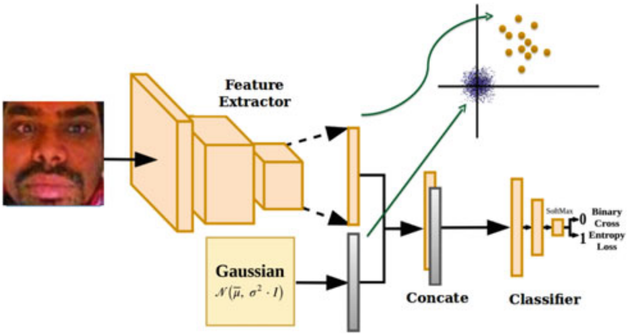


Fig. 12.7 Block diagram of the DCNN-based one-class classification approach proposed in [49]. Here, $\bar{\mu}$ and σ are mean and standard deviation parameters of a Gaussian, respectively, and **I** is the identity matrix

features are then fed into a classification network which is characterized by a fully connected neural network. The classification network assigns a confidence score for each feature representation. The output of the classification network is either 1 or 0. Here, 1 corresponds to the data sample belonging to the target class and 0 corresponds to the data sample belonging to the negative class. The entire network is trained end-to-end using binary cross-entropy loss. Extensive experiments were conducted, and it was demonstrated that this new DCNN-based one-class classification method achieved significant improvements over the recent state-of-the-art methods including one-class SVM, 1-MPM and support vector data description (SVDD) [49].

12.3.4 Quickest Intrusion Detection in Mobile AA

It is well known that a balance needs to be made between security and usability of a biometrics-based AA system [5, 50, 51]. In order to strike this balance in an AA scheme, following fundamental challenges should be factored.

1. Accuracy: How accurately does a mobile AA system detect an attacker or an intruder? Due to limitations of representation and classification models on mobile devices, behavioral and physiological biometrics-based methods do not provide good accuracy in practice [12, 52]. The AA system will be of little use in terms of security if it produces a high degree of false positives. On the other hand, a higher false negative rate would severely degrade the usability of the technology. Many recent approaches in the literature have attempted to address this factor by proposing better features and classifiers [12].

2. Latency: How long does it take to detect an attacker? If an AA system takes too long (e.g., 1–3 min) to detect an intrusion, it would grant an intruder plenty of time to extract sensitive information prior to the lock down. Hence, unless intruder

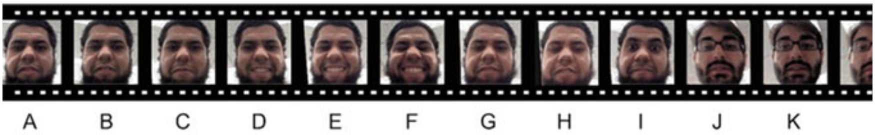


Fig. 12.8 Problem of quick intrusion detection in face-based AA systems. (A–I) show the genuine user with varying facial expressions. An intrusion occurs starting from (J). Active authentication systems should be able to detect intrusions as quickly as possible without causing too many false detections [41]

detection is sufficiently fast, the AA system would hold a little value in practice no matter how high its detection accuracy is.

Consider a series of observations captured from a front-facing camera of an Android device shown in Fig. 12.8. Frames (A–I) belong to the genuine user of the device. From frame J onward, an attacker starts to operate the device. In this scenario, frame J signifies a change point (i.e., an intrusion). The AA system should be able to detect intrusions with a minimal delay while maintaining a low rate of false detections. For instance, note the changes in genuine user's images in frames (D-F) due to camera orientation and facial expressions. While having a fast response, an AA system ideally should not falsely interpret these variations as intrusions.

3. Efficiency: How much resource does the system use? By definition, mobile AA systems are continuous processes that run as background applications. If they consume considerable amount of resources, memory, and processing power, it could slow down other applications and cause the battery to drain quickly. Despite the improvements in mobile memory and processors, battery capacity remains to be a constraint due to limitations in heat transfer and space [53]. Therefore, it can be expected to be the bottleneck in terms of efficiency in years to come. If an AA application causes battery to drain too quickly, then it is unrealistic to expect the users to use AA technology as they would typically opt out from using such applications [54]. Therefore, efficiency has a huge impact over the usability of AA as a technology. Recently, [38] studied the efficiency of a mobile AA system based on face biometric. Experiments were conducted on a Google Nexus 5 device with 2 GB of RAM and a quad-core 2.2 GHz CPU. It was shown that the normal usage of the device consumes about 520 mW of power and the facial attribute-based AA framework running at 4 frames per second consumes about 160.8 mW additional power. It is needless to say that nearly 30% increase in power consumption would take a toll on battery life. A trivial solution for this problem would be to decrease the sampling rate of data acquisition. However, the effects of such a measure on the detection performance have not been studied in the literature.

Many existing AA systems attempt to improve the accuracy of the system by proposing sophisticated features and classifiers. However, how fast an AA system could detect an intruder has not been widely studied in the literature. Yet, it remains to be an important feature of an AA system. In a recent paper [37, 41], authors addressed the problem of quickly detecting intrusions with lower false detection rates

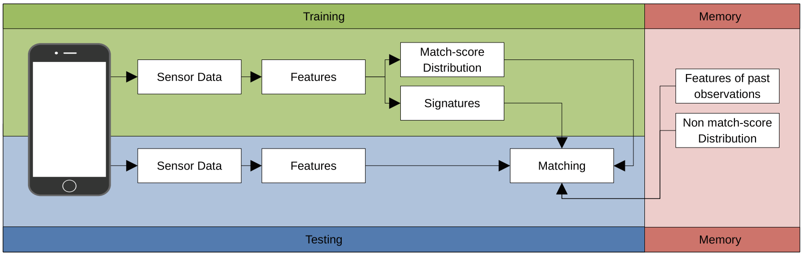


Fig. 12.9 An overview of the QCD-based AA method proposed in [41]

in mobile AA systems. They proposed Quickest Change Detection (QCD), which is a well-studied problem in statistical signal processing and information theory, for the purpose of intrusion detection in mobile AA systems. Figure 12.9 gives an overview of the proposed method. As opposed to a conventional AA system, this system utilizes all past observations along with distributions of match and non-match data of the genuine user to arrive at a decision. This proposed method does not require a specific feature nor a specific classifier; therefor, it can be built upon any existing AA system to enhance its performance. In particular, the introduced algorithms not only reduced the number of observations taken, but also improved the performance of the system in terms of latency and false detections. The validity of this result was demonstrated using various AA datasets.

12.3.5 Multi-user AA

Multiple-user active authentication [36, 40], in contrast with single-user active authentication, requires verification of identity of multiple subjects. Both traditional verification- and identification-based solutions fail to address the specific challenges presented in this problem. In a recent work [40], introduced Extremal Openset Rejection (EOR), a twofold mechanism with a sparse representation-based identification step and a verification step for this purpose. In the verification step, concentration of the sparsity vector and the overlap between matched and non-matched distributions are considered for decision making. Furthermore, a semi-parametric model based on EVT for modeling the distributions and an algorithm to estimate the parameters of extreme value distributions were also introduced in [40].

The EOR method essentially utilizes matched and non-matched distribution information on top of the identification criterion to make a better decision. It was shown that this additional processing has a significant gain particularly when identification criterion is poor (i.e., when a low number of users are enrolled). If a large number of classes are present, the additional verification step does not introduce a significant improvement. It was shown that EOR performs on par with the identification method in such scenarios. As a result, the EOR framework is particularly suited for multiple-user authentication problems.

Effectiveness of this method was demonstrated using three publicly available face-based mobile active authentication datasets. It was observed that verification-based algorithms generally performed well when low number of users were enrolled. On the one hand, identification-based algorithms required larger number of users to obtain good performance. However, good performance of both of these cases was confined to extremes with respect to number of users. On the other hand, the new EOR method yielded superior performance consistently as the number of users was varied. Hence, it was shown that EOR is suited for multiple AA in mobile devices where the number of users may vary.

12.4 AA Datasets

Data collection is one of the biggest challenges in mobile AA research. Several small-scale datasets are publicly available to the research community [12]. In particular, UMDAA-01 [22], MOBIO [21], and UMDAA-02 [55] are the three most commonly used face-based AA datasets. Sample images from these datasets are shown in Fig. 12.10. In what follows, we give a brief overview of these datasets.

The UMDAA-01 dataset [22] contains images captured using the front-facing camera of a iPhone 5S mobile device of 50 different individuals captured across three sessions with varying illumination conditions. Images of this dataset contain pose variations, occlusions, partial clippings as well as natural facial expressions as evident from the sample images shown in Fig. 12.10a.

The MOBIO dataset [21] contains videos of 152 subjects taken across two phases where each phase consists of six sessions. Videos in this dataset are acquired using a standard 2008 MacBook laptop computer and a NOKIA N93i mobile phone. Sample images from this dataset are shown in Fig. 12.10b.

The UMDAA-02 Dataset [55] is an unconstrained multimodal dataset where 18 sensor observations were recorded across a two-month period using a Nexus 5 mobile

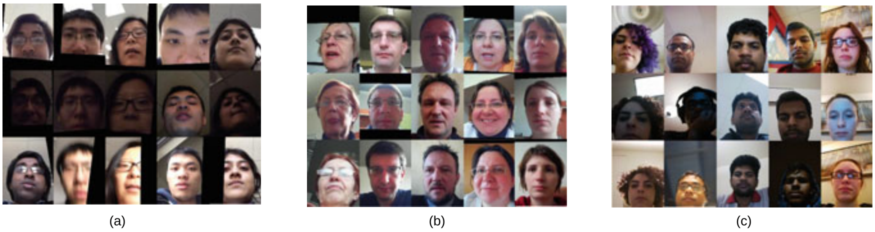


Fig. 12.10 Sample images from three face-based AA datasets. **a** UMDAA-01 [22], **b** MOBIO [21], **c** UMDAA-02 [55]. Each column represents sample images obtained for the same user

device. Unlike the earlier datasets, there exists a huge intra-class variation in this dataset in terms of poses, partial faces, illumination as well as appearances of the users as evident from the sample images shown in Fig. 12.10c.

12.5 Discussion

In this chapter, we provided a brief overview of recent advances in mobile-based active authentication methods. In particular, a special emphasis was given to the face-based methods. Continuous authentication on mobile devices promises to be an active area of research especially as more and more sensors are being added to the smartphone device and computation power of mobile devices has increased tremendously. There are, however, several challenges to be overcome before successfully designing a biometric-based continuous authentication system. Below, we list a few.

- A number of continuous authentication methods have been proposed in the literature that evaluate the performance of their proposed method on a variety of different datasets using different performance measures. However, there is no clear standard for evaluating the performance of different methods in the literature. Guidelines on an acceptable benchmark are needed.
- As discussed earlier, one of the major challenges in mobile-based AA is the datasets. Most mobile-based AA techniques discussed earlier have been evaluated on small- and mid-sized datasets consisting of hundreds of samples. However, in order to really see the significance and impact of various continuous authentication schemes in terms of usability and security, they need to be evaluated on large-scale datasets containing thousands and millions of samples.
- More usability and acceptability studies need to be conducted to really see the significance of AA in practice.

Acknowledgements This work was supported by US Office of Naval Research (ONR) Grant YIP N00014-16-1-3134.

References

1. Clarke N, Furnell S (2005) Authentication of users on mobile telephones: a survey of attitudes and practices. Comput Secur 24(7):519–527
2. Vance A (2010) If your password is 123456, just make it hackme (online; posted 20 Jan 2010). Available http://www.nytimes.com (online)
3. Aviv AJ, Gibson K, Mossop E, Blaze M, Smith JM (2010) Smudge attacks on smartphone touch screens. In: Proceedings of the 4th USENIX conference on offensive technologies, pp 1–7
4. Tapellini D (2014) Smart phone thefts rose to 3.1 million in 2013: industry solution falls short, while legislative efforts to curb theft continue (online; posted 28 May 2014). Available http://www.consumerreports.org/cro/news/2014/04/smart-phone-thefts-rose-to-3-1-million-last-year/index.htm (online)

5. Khan H, Hengartner U, Vogel D (2015) Usability and security perceptions of implicit authentication: convenient, secure, sometimes annoying. In: Eleventh symposium on usable privacy and security (SOUPS 2015), pp 225–239
6. Egelman S, Jain S, Portnoff RS, Liao K, Consolvo S, Wagner D (2014) Are you ready to lock? In: Proceedings of the 2014 ACM SIGSAC conference on computer and communications security, pp 750–761
7. Harbach M, von Zezschwitz E, Fichtner A, Luca AD, Smith M (2014) It's a hard lock life: a field study of smartphone (un)locking behavior and risk perception. In: Symposium on usable privacy and security (SOUPS 2014), pp 213–230
8. Guidorizzi RP (2013) Security: active authentication. IT Prof 15(4):4–7
9. Jakobsson M, Shi E, Golle P, Chow R (2009) Implicit authentication for mobile devices. In: Proceedings of USENIX
10. Shi E, Niu Y, Jakobsson M, Chow R (2011) Implicit authentication through learning user behavior. In: Proceedings of the 13th international conference on information security, pp 99–113
11. Clarke NL (2011) Transparent user authentication—biometrics. Springer, RFID and Behavioural Profiling
12. Patel VM, Chellappa R, Chandra D, Barbello B (2016) Continuous user authentication on mobile devices: recent progress and remaining challenges. IEEE Sig Process Mag 33(4):49–61
13. Frank M, Biedert R, Ma E, Martinovic I, Song D (2013) Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. IEEE Trans Inf Forensics Secur 8(1):136–148
14. Serwadda A, Phoha V, Wang Z (2013) Which verifiers work?: a benchmark evaluation of touch-based authentication algorithms. In: IEEE international conference on biometrics: theory. Sept, applications and systems, pp 1–8
15. Feng T, Liu Z, Kwon KA, Shi W, Carbunar B, Jiang Y, Nguyen N (2012) Continuous mobile authentication using touchscreen gestures. In: IEEE conference on technologies for homeland security, pp 451–456
16. Sherman M, Clark G, Yang Y, Sugrim S, Modig A, Lindqvist J, Oulasvirta A, Roos T (2014) User-generated free-form gestures for authentication: security and memorability. In: Proceedings of the 12th annual international conference on mobile systems, applications, and services, pp 176–189
17. Zhao X, Feng T, Shi W, Kakadiaris I (2014) Mobile user authentication using statistical touch dynamics images. IEEE Trans Inf Forensics Secur 9(11):1780–1789
18. Zhang H, Patel VM, Fathy ME, Chellappa R (2015) Touch gesture-based active user authentication using dictionaries. In: IEEE winter conference on applications of computer vision
19. Bo C, Zhang L, Li XY, Huang Q, Wang Y (2013) Silentsense: silent user identification via touch and movement behavioral biometrics. In: Proceedings of the 19th annual international conference on mobile computing & networking, ser. MobiCom '13. ACM, New York, NY, USA, pp 187–190
20. Fridman L, Weber S, Greenstadt R, Kam M (2015) Active authentication on mobile devices via stylometry, GPS location, web browsing behavior, and application usage patterns. IEEE Syst J
21. McCool C, Marcel S, Hadid A, Pietikainen M, Matejka P, Cernocky J, Poh N, Kittler J, Larcher A, Levy C, Matrouf D, Bonastre JF, Tresadern P, Cootes T (2012) Bi-modal person recognition on a mobile phone: using mobile phone data. In: IEEE international conference on multimedia and expo workshops, pp 635–640
22. Fathy ME, Patel VM, Chellappa R (2015) Face-based active authentication on mobile devices. In: IEEE international conference on acoustics, speech and signal processing
23. Samangouei P, Patel VM, Chellappa R (2015) Attribute-based continuous user authentication on mobile devices. In: IEEE international conference on biometrics: theory, applications and systems
24. Hadid A, Heikkila J, Silven O, Pietikainen M (2007) Face and eye detection for person authentication in mobile phones. In: ACM/IEEE international conference on distributed smart cameras, pp 101–108

25. Derawi M, Nickel C, Bours P, Busch C (2010) Unobtrusive user-authentication on mobile phones using biometric gait recognition. In: International conference on intelligent information hiding and multimedia signal processing, pp 306–311
26. Primo A, Phoha V, Kumar R, Serwadda A (2014) Context-aware active authentication using smartphone accelerometer measurements. In: IEEE conference on computer vision and pattern recognition workshops, pp 98–105
27. Crouse D, Han H, Chandra D, Barbello B, Jain AK (2015) Continuous authentication of mobile user: fusion of face image and inertial measurement unit data. In: International conference on biometrics
28. Zhang H, Patel VM, Chellappa R (2015) Robust multimodal recognition via multitask multivariate low-rank representations. In: IEEE international conference on automatic face and gesture recognition, vol 1, pp 1–8
29. Zhang H, Patel VM, Shekhar S, Chellappa R (2015) Domain adaptive sparse representation-based classification. In IEEE international conference on automatic face and gesture recognition, vol 1, pp 1–8
30. Abramson M, Aha DW (2013) User authentication from web browsing behavior. In: Florida artificial intelligence research society conference. AAAI Press
31. Li F, Clarke N, Papadaki M, Dowland P (2014) Active authentication for mobile devices utilising behaviour profiling. Int J Inf Secur 13(3):229–244
32. Saevanee H, Clarke N, Furnell S, Biscione V (2014) Text-based active authentication for mobile devices. In: Cuppens-Boulahia N, Cuppens F, Jajodia S, Abou El Kalam A, Sans T (eds) ICT systems security and privacy protection, ser. IFIP advances in information and communication technology. Springer, Berlin, Heidelberg, vol 428, pp 99–112
33. Gascon H, Uellenbeck S, Wolf C, Rieck K (2014) Continuous authentication on mobile devices by analysis of typing motion behavior. Sicherheit 2014:1–12
34. Holz C, Buthpitiya S, Knaust M (2015) Bodyprint: biometric user identification on mobile devices using the capacitive touchscreen to scan body parts. In: Proceedings of the 33rd annual ACM conference on human factors in computing systems. ACM, New York, NY, USA, pp 3011–3014
35. Perera P, Patel VM (2017) Extreme value analysis for mobile active user authentication. In: IEEE international conference on automatic face and gesture recognition
36. Perera P, Patel VM (2017) Towards multiple user active authentication in mobile devices. In: IEEE international conference on automatic face and gesture recognition
37. Perera P, Patel VM (2016) Quickest intrusion detection in mobile active user authentication. In: International conference on biometrics theory, applications and systems
38. Samangouei P, Patel VM, Chellappa R (2016) Facial attributes for active authentication on mobile devices. Image Vis Comput 58:181–192
39. Samangouei P, Chellappa R (2016) Convolutional neural networks for attribute-based active authentication on mobile devices. In: IEEE international conference on biometrics: theory, applications, and systems
40. Perera P, Patel VM (2018) Facebased multiple user active authentication on mobile devices. IEEE Trans Inf Forensics Secur 14:1240–1250
41. Perera P, Patel VM (2018) Efficient and low latency detection of intruders in mobile active authentication. IEEE Trans Inf Forensics Secur 13(6):1392–1405
42. Sarkar S, Patel VM, Chellappa R (2016) Deep feature-based face detection on mobile devices. In: IEEE international conference on identity, security and behavior analysis
43. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, p 2012
44. Mahbub U, Patel VM, Chandra D, Barbello B, Chellappa R (2016) Partial face detection for continuous authentication. In: IEEE international conference on image processing
45. Abeni P, Baltatu M, D'Alessandro R (2006) Nis03-4: implementing biometrics-based authentication for mobile devices. In: IEEE global telecommunications conference, pp 1–5
46. Viola PA, Jones MJ (2004) Robust real-time face detection. Int J Comput Vis 57(2):137–154

47. Kumar N, Berg A, Belhumeur P, Nayar S (2011) Describable visual attributes for face verification and image search. IEEE Trans Pattern Anal Mach Intell 33(10):1962–1977
48. Perera P, Patel VM (2018) Dual-minimax probability machines for one-class mobile active authentication. In: IEEE international conference on biometrics: theory, applications, and systems
49. Oza PB, Patel VM (2018) One-class convolutional neural network. IEEE Sig Process Lett 26:277–281
50. Clarke N, Karatzouni S, Furnell S (2009) Flexible and Transparent User Authentication for Mobile Devices. In: Emerging challenges for security, privacy and trust: 24th IFIP TC 11 international information security conference, SEC 2009, Pafos, Cyprus, 18–20 May 2009. Proceedings. Springer, Berlin, Heidelberg, pp 1–12
51. Crawford H, Renaud K (2014) Understanding user perceptions of transparent authentication on a mobile device. J Trust Manag 1(7):1–28
52. Meng W, Wong DS, Furnell S, Zhou J (2015) Surveying the development of biometric user authentication on mobile phones. IEEE Commun Surv Tutorials 17(3):1268–1293
53. Li JGWD, Hao S, Halfond GJ (2014) An empirical study of the energy consumption of android applications. In: IEEE international conference on software maintenance and evolution (ICSME)
54. Lee W (2013) Mobile apps and power consumption—basics, part 1. Available https://developer.qualcomm.com/blog/mobile-apps-and-power-consumption-basics-part-1 (online)
55. Mahbub U, Sakar S, Patel V, Chellappa R (2016) Active authentication for smartphones: a challenge data set and benchmark results. In: IEEE international conference on biometrics: theory applications and systems

Chapter 13
Mobile User Re-authentication Using Clothing Information

Hoang (Mark) Nguyen, Ajita Rattani and Reza Derakhshani

Abstract Biometric authentication has become a popular alternative to passwords on mobile devices. However, most implementations do not incorporate any mechanisms to ascertain whether the originally authenticated user is still in control of the mobile device. Thus, the user has to re-scan for any subsequent device access, which may lead to biometric scan fatigue. One solution to this problem is to re-authenticate the user via ancillary surrogates of identity that are likely to be stable and unique in the short term and easier to acquire compared to the primary biometric modality, such as opportunistically captured clothing information. The aim of this paper is to investigate such clothing information as a soft biometrics for short-term mobile user re-authentication. To this aim, we propose a novel method for segmentation and matching of clothing ROI from images captured via front-facing camera of mobile devices, without explicitly requiring the face to be present. Experimental investigations on a large-scale mobile dataset show error rates as low as 2.5% using this method.

13.1 Introduction

Mobile devices are playing a significant role in daily life, not only for communications but also for entertainment, e-commerce, and even remote health services. However, mobile phones are misplaced, lost, and stolen more than other computing devices. Therefore, efforts have been directed at the development of biometrically secure

H. (Mark) Nguyen (✉) · R. Derakhshani
Department of Computer Science and Electrical Engineering,
University of Missouri at Kansas City, Kansas City, USA
e-mail: hdnf39@mail.umkc.edu

R. Derakhshani
e-mail: derakhshanir@umkc.edu

A. Rattani
Department of Electrical Engineering and Computer Science,
Wichita State University, Wichita, KS, USA
e-mail: ajita.rattani@wichita.edu

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision
and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_13

mobile access and transactions. The use of biometric technology in mobile devices is referred to as *mobile biometrics* [9, 16, 17, 24]. Biometrics Research Group, Inc.[1] has predicted that by 2020, mobile biometrics will transition from consumer adoption phase to full maturity, enabling the technology to overtake existing authentication technologies. By 2020, it is estimated that biometrics will be ubiquitous, installed on 100 percent of mobile devices.

Thus, many commercial solutions as well as well as academic studies have been focusing on mobile user authentication via strong primary biometric traits. In particular, modalities based on face [9, 16] and ocular region [14, 15, 17, 20] acquired using selfie images are of interest given that they do not require any specialized sensors.[2] Fingerprint and near-infrared iris captured [4, 25] using dedicated sensors installed in mobile devices have also been used for mobile user authentication.

However, most of these methods focus on entering the user into the authenticated state via the primary biometric but provide no explicit or robust solution to keep the user in that state. In other words, they have no mechanism to determine whether the user authorized after the initial successful authentication is still the same person in control of the device [10]. If the device locks up or logs out after the initial access, the user has to frequently re-scan his or her biometrics using the primary modality to regain access to the device and its services, each time requiring a certain level of cooperation and attention, leading to bad user experience. Alternatively, if a timer is used to extend the initial authenticated state, there is still a risk of illegitimate access to the sensitive information on the device by an intruder if the device was taken from its original user in the meantime. To mitigate this problem, there is a need for short-term, low friction user re-authentication to properly extend the authenticated state after the initial primary biometric scan by the authorized user [2, 10, 23, 26].

Two most important factors for frequent and even continuous user authentication are reliability and usability. Primary biometrics such as face, eye, and finger scans are highly reliable but require a non-negligible active user cooperation for an acceptable scan (e.g., aligning the face or eyes with the camera or placing a clean finger on the fingerprint scanner), reducing their utility for frequent re-authentication. Further, these traits might not be available due to the user's pose. Less cooperative soft biometrics such as gender, skin color, and other face attributes, as well as other modalities like keystrokes and device movement dynamics [12, 23] have gained attention for user re-authentication in the background.

In this work, we investigate the use of clothing information as soft biometrics for short-term mobile user re-authentication. Clothing information has been studied extensively in person re-identification for multi-camera surveillance systems [5–7]. The advantages of using clothing information for mobile user re-authentication are as follows:

[1] https://www.biometricupdate.com/201703/special-report-mobile-biometric-applications.

[2] A selfie is a self-portrait image of a user captured using the ubiquitous front-facing cameras available in virtually all mobile devices.

- Clothing, as something that one has, and after being temporarily tied to the user identity at the time of the primary biometric scan, is usually unique and stable enough to be used for re-authentication for ensuing several minutes.
- Though clothing, as detailed above, may constitute a temporary visual representation of an individual, it is inherently revocable and unlike other soft biometrics the information stored in the template generally does not compromise user's privacy.
- Clothing ROI is a much larger target compared to the face and eyes, and thus it can be acquired from the front-facing camera while a user is naturally interacting with the target application with no explicit cooperation (except an initial consent to allow the method).

It should be noted though that this method is not applicable to scenarios where people wear uniform clothing, nor when the device camera is not in the general direction of the user's torso. The latter is indeed a benefit, since re-authentication should not happen when the user is not naturally interacting with the app that requested the service. That is also the time window when the OS permissions allow the use of the device cameras.

Our earlier study in [11] consisted of a preliminary investigation on the use of clothing information for mobile user re-authentication. The new contribution of this work over [11] are as follows:

1. A new deep learning-based method for more accurate segmentation of clothing ROI from selfie images that is robust to different user poses, rendering this method much more applicable to everyday mobile use cases.
2. Evaluation of SURF keypoint detectors and patch descriptors for matching clothing ROIs from selfie image pairs, followed by a comparative evaluation of this non-learning-based texture descriptor method with learning-based methods across various scales to better understand the pros and cons of each methodology.

The rest of this paper is organized as follows: Sect. 13.2 describes the existing work related to continuous mobile user authentication. Section 13.3 describes the proposed segmentation and matching methods for clothing-based short-term user re-authentication. Experimental validations of the proposed method are discussed in Sect. 13.4. Conclusions and future work are given in Sect. 13.5.

13.2 Previous Work

In this section, we discuss existing soft biometric methods applicable to the mobile device user re-authentication.

Samangouei et al. [23] proposed facial attributes such as gender, ethnicity, eyeglasses, hair color, skin type, and face shape as an auxiliary authentication method for mobile devices. Binary SVM classifiers were trained for each attribute. The learned classifiers were applied to the selfie image of the user for attribute's extraction. Authentication was done by comparing the extracted attributes with the enrolled attributes of the user.

Zhao et al. [26] investigated the touch-based continuous mobile authentication via proposing a novel Graphic Touch Gesture Feature (GTGF). In this method, touch traces were converted to images for the explicit representation of the touch dynamics. The touch sequences were first segmented and normalized so that traces have a fixed number of sample points. Then, the samples on the normalized traces were converted into shapes and intensity values of the GTGF. User authentication was performed by computing L1-norm between a pair of GTGF images. In [22] a text-based multimodal biometric approach utilizing linguistic analysis, keystroke dynamics, and behavioral profiling was proposed for continuous mobile user authentication.

Crouse et al. [2] proposed an unobtrusive continuous authentication system based on face matching. Performance and accuracy for unconstrained face matching were improved by integrating data from the device accelerometer, gyroscope, and magnetometer to correct camera sensor orientation and hence face image.

Rattani et al. [18, 19] proposed convolutional neural networks for gender and age prediction from ocular images captured using mobile devices for performance enhancement and potential re-authentication. In another work [10], authors exploited the use of eyebrows for short-term mobile user authentication. Eyebrows, being one-sixth of the facial region, is computationally efficient and offers fast throughput for continuous re-authentication in mobile devices. To this aim, the histogram of oriented gradients and GIST descriptors extracted from left and right eyebrow regions were evaluated.

The above studies, though helpful in their given contexts, do not solve the problem of user re-authentication without needing the face to be in view, or they may require user interaction with an additional touch-based modality. To the best of our knowledge, the line of studies starting with [11] was the first attempt on continuous user authentication using clothing information from selfie images in the mobile environment. In that the preliminary study, learning-based methods using local texture descriptors along with support vector machines (SVM) were applied on clothing ROI that was approximated through heuristics.

13.3 Proposed Method

The main steps involved in the proposed method are (a) selfie-pose-invariant clothing ROI segmentation and (b) robust matching of the features extracted from clothing ROI. We evaluated the efficacy of both learning and non-learning methods for the latter. Next, we discuss these steps in detail.

13.3.1 Clothing Segmentation

The segmentation task can be viewed as a pixel-wise labeling where the system differentiates between the pixels of clothing from those of the background. Deep learning-based segmentation methods have been outperforming traditional methods. It has become common to use convolutional encoder–decoder models for this purpose. The encoder layers extract features from input data while the decoder layers reconstruct the image by way of the feature maps [8]. The model produces a binary mask of the original image size delineating the background from the foreground target object, respectively.

In this work, we used U-Net [21]-based deep learning model for clothing ROI segmentation. U-Net is a convolutional neural network that was originally developed for biomedical image segmentation. The network architecture of U-Net consists of contracting part (encoder) on the left and expansive path (decoder) on the right. The encoder is a repeated application of two 3 × 3 convolutions, followed by rectified linear units (ReLU), and 2 × 2 max pooling operation. Similarly, each decoder layer consists of an upsampling using 2 × 2 up-convolution, a concatenation of corresponding feature maps from the contracting path, and two 3 × 3 convolutions followed by ReLUs. This was also the first network to introduce skip connections for directly connecting the upsampling and downsampling layers. This allows the network to take the context of the image into account, which could be lost through the convolution operation otherwise. The architecture of the network is designed for parametrization with fewer training images, and it yields more precise segmentations.

For clothing segmentation, we trained the U-Net model with 1000 selfie images collected from the web. The dataset was further augmented by adding Gaussian blur, scaling, and rotation to the original selfie images, along with target binary masks. The training clothing masks were created using MATLAB’s built-in “imageLabeler” function.

Figure 13.1 shows the architecture of U-Net for clothing mask generation from selfie images.

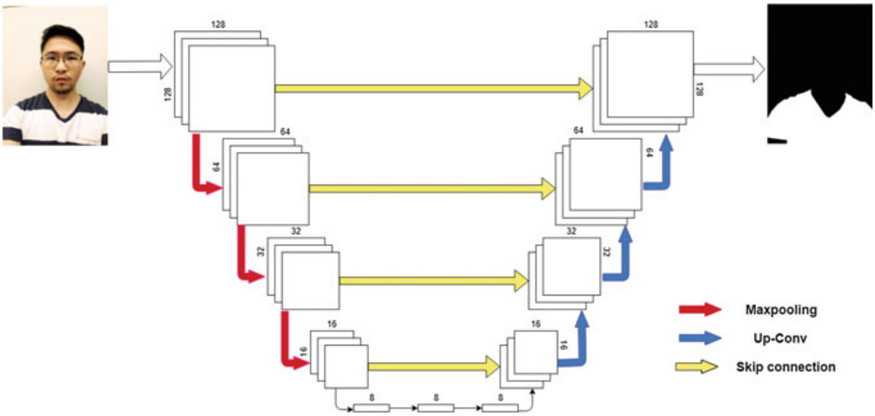


Fig. 13.1 Architecture of U-Net model used for clothing mask generation from selfie images

13.3.2 Clothing Matching

Clothing matching is the process of confirming whether two visual representations are from the same clothes or not. This is done by feature extraction from segmented clothing ROIs and matching them using either learning or non-learning-based methods. Next, we discuss our proposed learning and non-learning methods for the purpose.

13.3.2.1 Learning-Based Method

We define a learning-based method as one where the discriminant (or the similarity metric) is learned via training data. In the proposed learning-based method, tile texture features are used to train an SVM as the learned similarity metric. The trained SVM is then used for re-authentication. Based on the literature features and our various experiments, we found local binary pattern (LBP) [13], histogram of oriented gradient (HOG) [3], and color histogram (CH) to be most effective for this task. LBP is a simple visual descriptor that encodes the differences between the given center pixel with those in its neighborhood. HOG computes the local gradient orientation of the dense grid with local contrast normalization. LBP and HOG both operate on gray-scale images. CH generates color information from the histogram of R, G, and B channels. All features are extracted by dividing clothing ROI into 2 × 3 non-overlapping tiles at four different image scales (1×, 0.5×, 0.25× and 0.125×), an arrangement that was experimentally determined to be most effective. All these LBP, HOG, and CH feature vectors are then concatenated into a single vector as shown in Fig. 13.2 and used for training and testing the SVMs. We experimentally determined linear SVMs to provide the best generalization.

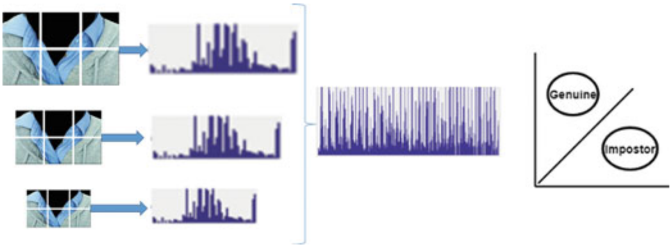


Fig. 13.2 Features extracted from a clothing ROI that is divided into 2 × 3 blocks at three different scales. All the extracted features from the different scales are concatenated into a single vector prior to classification

13.3.2.2 Non-learning-Based Method

We define a non-learning-based method as one where the discriminant is a pre-defined distance metric, such as Euclidean or Manhattan distance. In our non-learning-based method, we used the venerable speeded up robust features (SURF) [1]. SURF has been proven to be one of the best local feature detectors and descriptors for object recognition and image classification. In order to detect interest points, it uses Hessian matrix with the approximation of Gaussian smoothing. Similar to the scale-invariant feature transform (SIFT), interest points are calculated at different scales of the image pyramid. The descriptors around each interest point are computed using the first-order Haar wavelet responses which represent the intensity distribution of pixels within a block. The match score is computed as the number of matched SURF points between enrollment and verification clothing ROIs using the sum of absolute differences (Manhattan distance), experimentally deemed to be the best for this use case. Figure 13.3 shows the matching of SURF descriptors from clothing pairs coming from same (genuine) and different (impostor) clothing ROIs.

The obvious advantage of learning-based method is its higher accuracy over non-learning methods given its data-driven similarity metric. However, non-learning methods are usually computationally more efficient, do not require an extensive training process, and being more generic they may generalize better over certain unseen datasets. Figure 13.4 shows the overall proposed system.

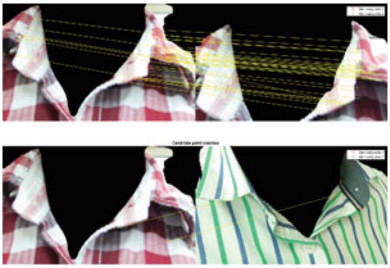


Fig. 13.3 SURF point matching between a pair of similar (genuine) clothing ROIs (top) and different (impostor) clothing (below)

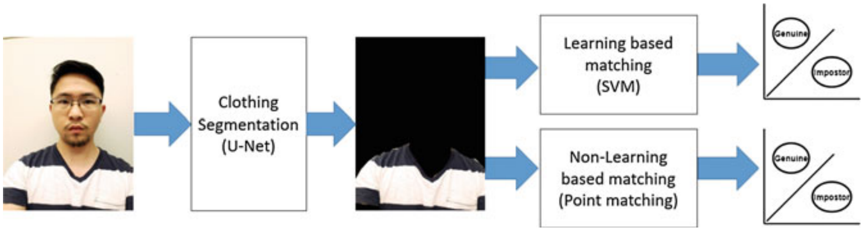


Fig. 13.4 Overview of the short-term user re-authentication system based on clothing information. The main steps are clothing segmentation using U-Net followed by matching using proposed learning or non-learning-based methods

13.4 Experimental Validation

13.4.1 Dataset and Protocol

The dataset used in this work is a subset of full face mobile dataset used to generate VISOB dataset [17]. VISOB dataset was collected by acquiring full face selfie images from around 550 healthy adults using front-facing cameras of mobile devices. The subset of the dataset consisting of about 240,000 selfie images from 293 subjects using an OPPO N1 cellular phone. Out of the whole subset, the pre-trained segmentation algorithm detected masks for about 85,000 of images containing enough clothing information. Approximately half of these images were used for training and testing. Both the sets were further subdivided based on lighting conditions at the time of capture: daylight and indoor office lighting, for experimental analysis of system performance across different lighting conditions. Equal error rate (EER), area under the ROC curve (AUC), and precision and recall were used as performance metrics in our analysis.

13.4.2 Results

In this section, we present and discuss the result of proposed clothing segmentation and matching using learning and non-learning-based methods.

13.4.2.1 Clothing Segmentation

In order to evaluate the segmentation accuracy, we used precision and recall metrics given in Eqs. 13.1 and 13.2, respectively. In these equations, S is the segmentation mask obtained by U-Net model, and R is the ground truth label mask. Precision is the fraction of pixels that are segmented correctly over the total pixels in clothing mask generated by U-Net. Recall is the fraction of pixels that are segmented correctly over the total pixels in the ground truth label mask. Using the above equations, we obtained precision and recall of 94.73 and 94.03%, respectively. The high precision and recall rates suggest the efficacy of the proposed method for clothing ROI segmentation. Figure 13.5 shows the examples of segmented clothes and clothing masks.

$$Precision = \frac{S \cap R}{|S|} \tag{13.1}$$

$$Recall = \frac{S \cap R}{|R|} \tag{13.2}$$

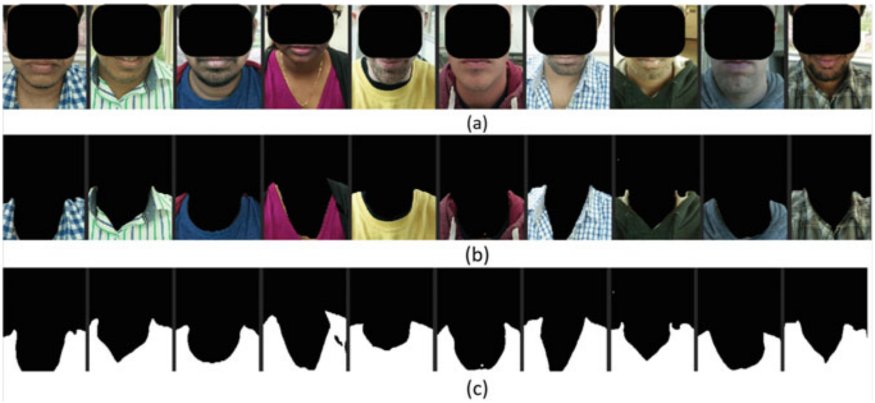


Fig. 13.5 Example of **a** original selfie images, **b** segmented clothes, **c** and the corresponding masks obtained by our U-Net model segmentation. The eye regions have been masked in order to preserve the privacy of the participants

13.4.2.2 Learning-Based Clothing Matching

Table 13.1 shows the performance of learning-based method for clothing matching in terms of EER and AUC across same and different lighting conditions. Recall that learning-based method consists of feature level fusion of LBP, HOG, and CH feature vectors for SVM training and classification. Understandably, a very low error rate is obtained when training and testing sets are acquired under the same lighting conditions. The least EER of 2.5% was obtained when training and testing sets were acquired using indoor office lighting condition. However, the EER increased when lighting conditions were varied. The EER increased to 10.7% when the training images were acquired in office lighting conditions and test images came from daylight captures. Similarly, EER increased to 13.9% when the training images were acquired under daylight conditions and test images came from indoor office lighting conditions. This suggests that the method is sensitive to illumination variations. Figures 13.6 and 13.7 show ROC curves of the learning-based method across same and different lighting conditions.

Table 13.1 AUCs and EERs of learning-based method with same and different lighting conditions

Train	Test	AUC	EER (%)
Office light	Office light	0.994	2.5
Daylight	Daylight	0.992	3.5
Office light	Daylight	0.954	10.7
Daylight	Office light	0.937	13.9

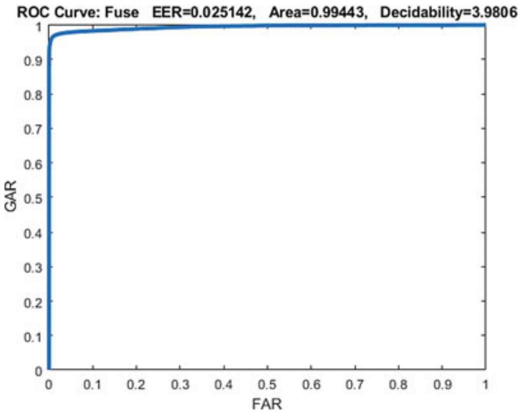


Fig. 13.6 ROC of learning-based method for clothing matching when the training and test images are all acquired under indoor office lighting conditions

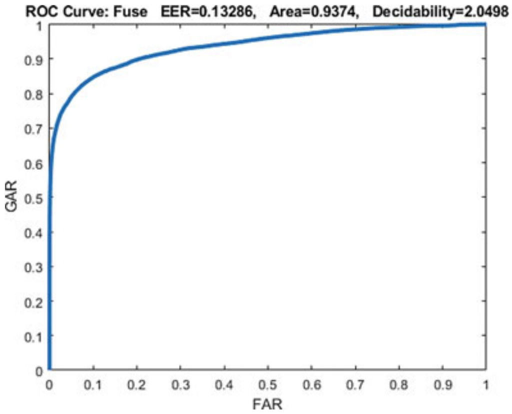


Fig. 13.7 ROC of learning-based method when the training and test images are acquired under daylight and office lighting conditions, respectively

13.4.2.3 Non-learning-Based Clothing Matching

Table 13.2 shows the performance of non-learning-based SURF matcher. Again, it can be seen that lower EERs are obtained when the pair of selfie images were captured under the same lighting conditions. EERs of 11.9 and 13.9% were obtained when images were acquired under the same office lighting or daylight conditions,

Table 13.2 AUCs and EERs of non-learning method using same and different lighting conditions

Train	Test	AUC	EER (%)
Office light	Office light	0.943	11.9
Daylight	Daylight	0.929	13.8
Office light	Daylight	0.888	19.7
Daylight	Office light	0.875	18.9

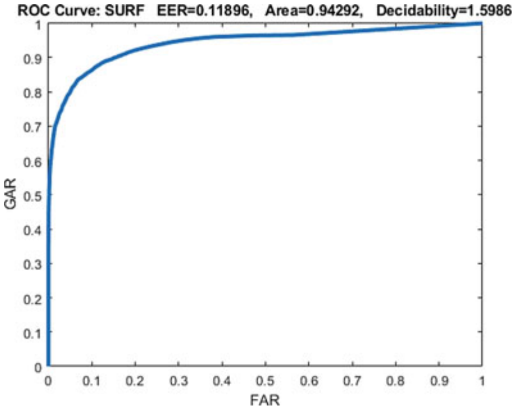


Fig. 13.8 ROC of the non-learning method when the training and test images are acquired under office lighting condition

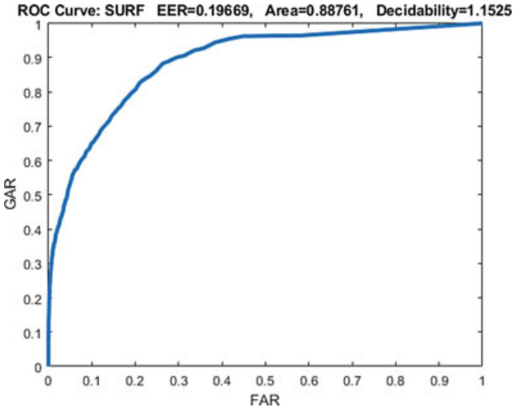


Fig. 13.9 ROC of the non-learning method when the training and testing images are acquired under office and daylight conditions, respectively

respectively. However, the performance drops for training and testing across different lighting conditions. 18.9 and 19.7% EERs were obtained when training and testing images were acquired under mixed office light and daylight conditions.

Figures 13.8 and 13.9 show the ROCs for non-learning clothing matching under same and different lighting conditions, respectively.

13.5 Conclusion and Future Work

In this paper, we showed the utility of partial clothing information, seen on the user's upper torso during uncooperative, free form interaction with a mobile device with front-facing cameras, for short-term re-authentication. We treat such clothing information as a soft identifier (something that user has and does not change in short term) if and when tied to a strong identifier such as a primary biometric that enters the user into the authenticated state. Here we show that, using our proposed clothing segmentation and matching methods, one can obtain acceptable error rates to keep the user authenticated if he/she returns to a previously (biometrically) authorized device after a short period of time, without needing extra explicit biometric scans, for better user experience. The obtained error rates for matching clothing information are quite low when the verification clothing images are captured under similar lighting conditions that were used for training (2.5 and 11.9% EERs for learning and non-learning-based matching methods, respectively). However, the error rates increase across different lighting conditions. As a part of future work, a large-scale retraining and evaluation of the proposed methods will be conducted on other available mobile datasets. The proposed methods can be made more resilient to varying lighting conditions by including lighting variability into larger training sets, utilizing lighting-equalizing preprocessing, and by employing more resilient matching. More specifically, deep learning-based methods will be developed for matching clothing ROIs. Further, an adaptive fusion of clothing information with other available soft biometrics traits, such as the presence of eyeglasses, skin color, and gender, will be investigated for further performance enhancements.

References

1. Bay H, Tuytelaars T, Van Gool L (2006) Surf: speeded up robust features. In: Leonardis A, Bischof H, Pinz A (eds) Computer Vision - ECCV 2006. Springer, Berlin Heidelberg, Berlin, Heidelberg, pp 404–417
2. Crouse D, Han H, Chandra D, Barbello B, Jain AK (2015) Continuous authentication of mobile user: fusion of face image and inertial measurement unit data. In: International conference on biometrics (ICB), pp 135–142
3. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol 1, pp 886–893
4. Gao M, Hu X, Cao B, Li D (2014) Fingerprint sensors in mobile devices. In: 2014 9th IEEE conference on industrial electronics and applications, pp 1437–1440
5. Hermans A, Beyer L, Leibe B (2017) In defense of the triplet loss for person re-identification. CoRR, abs/1703.07737
6. Jaha ES, Nixon MS (2014) Soft biometrics for subject identification using clothing attributes. In: IEEE international joint conference on biometrics, pp 1–6
7. Jaha ES, Nixon MS (2016) From clothing to identity: Manual and automatic soft biometrics. IEEE Trans Inf Forensics Secur 11(10):2377–2390
8. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: 2015 IEEE conference on computer vision and pattern recognition (CVPR), pp 3431–3440
9. Marsico MD, Galdi C, Nappi M, Riccio D (2014) Firme: face and iris recognition for mobile engagement. Image Vis Comput 32(12):1161–1172
10. Mohammad AS, Rattani A, Derakhshani R (2018) Short-term user authentication using eyebrows biometric for smartphone devices. In: IEEE computer science and electronic engineering conference
11. Nguyen H, Sai R, Li Z, Derakhshani R (2018) User re-identification using clothing information for smartphones (accepted). In: IEEE International symposium on technologies for homeland security (HST), pp 1–6
12. Niinuma K, Park U, Jain AK (2010) Soft biometric traits for continuous user authentication [dec 10 771–780]. IEEE Trans Inf Forensics Secur 6(4):771–780

13. Ojala T, Pietikainen M, Maenpaa T (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans Pattern Anal Mach Intell 24(7):971–987
14. Rattani A, Derakhshani R (2017) On fine-tuning convolutional neural networks for smartphone based ocular recognition. In: 2017 IEEE international joint conference on biometrics (IJCB), pp 762–767
15. Rattani A, Derakhshani R (2017) Online co-training in mobile ocular biometric recognition. In: 2017 IEEE international symposium on technologies for homeland security (HST), pp 1–5
16. Rattani A, Derakhshani R (2018) A survey of mobile face biometrics. Comput Electr Eng 72:39–52
17. Rattani A, Derakhshani R, Saripalle SK, Gottemukkula V (2016) Icip 2016 competition on mobile ocular biometric recognition. In: 2016 IEEE international conference on image processing (ICIP), pp 320–324
18. Rattani A, Reddy N, Derakhshani R (2017) Convolutional neural network for age classification from smart-phone based ocular images. In: 2017 IEEE international joint conference on biometrics (IJCB), pp 756–761
19. Rattani A, Reddy N, Derakhshani R (2018) Convolutional neural networks for gender prediction from smartphone-based ocular images. IET Biom 7:423–430(7)
20. Reddy N, Rattani A, Derakhshani R (2018) Ocularnet: deep patch-based ocular biometric recognition. In: 2018 IEEE international symposium on technologies for homeland security (HST), pp 1–6
21. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. CoRR abs/1505.04597
22. Saevanee H, Clarke N, Furnell S, Biscione V (2015) Continuous user authentication using multi-modal biometrics. Comput Secur 53:234–246
23. Samangouei P, Patel VM, Chellappa R (2015) Attribute-based continuous user authentication on mobile devices. In: 2015 IEEE 7th international conference on biometrics theory, applications and systems (BTAS), pp 1–8
24. Tao Q, Veldhuis R (2006) Biometric authentication for a mobile personal device. Third annual international conference on mobile and ubiquitous systems: networking services., San Jose CA, pp 1–3
25. Thavalengal S, Bigioi P, Corcoran P (2015) Iris authentication in handheld devices - considerations for constraint-free acquisition. IEEE Trans Consum Electron 61(2):245–253
26. Zhao X, Feng T, Shi W (2013) Continuous mobile authentication using a novel graphic touch gesture feature. In: 2013 IEEE sixth international conference on biometrics: theory, applications and systems (BTAS), pp 1–6

Part IV
Security, Privacy, Usability and Protocol for Selfie Biometrics

Chapter 14
A Framework for Secure Selfie-Based Biometric Authentication in the Cloud

Veeru Talreja, Terry Ferrett, Matthew C. Valenti and Arun Ross

Abstract Cloud-based selfie authentication has multiple advantages over on-device selfie authentication: Cloud-based authentication can support nomadic access from multiple devices including those not owned by the user, can leverage cheap and scalable utility computing, and can enable rapid innovation by allowing new matching algorithms to be continually deployed with no need to update the local device. This chapter presents a framework for a cloud-based selfie biometric authentication, which is termed *Selfie-Biometrics-as-a-Service* (SBaaS). By leveraging Platform-as-a-Service (PaaS) concepts, the framework is designed to enable independent software vendors to develop extensions and add-ons to a provider's core application. In particular, the framework creates an innovative marketplace for biometric algorithms by providing a standard pre-built interface for the development and submission of new matching algorithms. When an authentication request is submitted, a criteria is used to select an appropriate matching algorithm. Every time a particular algorithm is selected, the corresponding developer is rendered a micropayment. Also presented in this chapter are solutions for preserving the confidentiality of biometrics stored in the cloud. This can be achieved through the use of biocryptosystems, which are secure biometric architectures involving the conversion of biometric features into secure signals that can be stored in the biometric database but are still useable for authentication. To provide a concrete example, a case study of a selfie-based ocular recognition system is disclosed, and detailed descriptions are provided of the user and developer interfaces.

V. Talreja · T. Ferrett · M. C. Valenti
West Virginia University, Morgantown, WV, USA
e-mail: vtalreja@mix.wvu.edu

T. Ferrett
e-mail: terry.ferrett@mail.wvu.edu

M. C. Valenti
e-mail: matthew.valenti@mail.wvu.edu

A. Ross (✉)
Michigan State University, East Lansing, MI, USA
e-mail: rossarun@cse.msu.edu

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision
and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_14

14.1 Introduction

Recently, there has been tremendous interest in incorporating selfie biometric solutions into consumer electronic such as smartphones and tablets. Traditionally, specialized biometric sensors have been proprietary with high cost, low market adoption, and limited compatibility with competing systems [28]. However, smartphones are equipped with cameras and other sensors suitable for biometric sensing tasks in the context of face, fingerprint, ocular, and gait recognition. The presence of high-resolution front-facing cameras, in particular, offers the possibility of performing selfie biometric recognition within the confines of the device.

Selfie biometric authentication is currently being used by e-commerce companies to enable customers to purchase merchandise more conveniently, by using a selfie for login authentication and payment confirmation. It is also being used by banks and other organizations to protect the security of financial records and access to funds. Major organizations that have begun using a selfie authentication technology include Amazon, Mastercard, and Alibaba.

When selfie biometrics are used to authenticate a user on a personal device, the enrollment is usually local to the device and successful authentication unlocks a locally stored key for use in a conventional private-key cryptosystem. However, performing selfie biometric authentication using only the resources local to the device is challenging due to several confounding factors including variations in head pose, ambient illumination, facial expression, occlusion, and limited availability of resources within the smartphone for storage and computation [11, 13]. Another significant challenge that cannot be met by a solitary personal device is supporting the nomadic usage habits demanded by today's consumers, who want to be able to gain access to services from any location using any number of personal devices or from public infrastructure such as ATMs and pay stations. As an alternative to confining authentication to the local device, biometric authentication can be performed in the cloud. In this regard, *cloud computing* may be harnessed as a viable option. Cloud computing [19] facilitates the outsourcing of computing and storage tasks to infrastructures managed by dedicated providers—providing an opportunity to surpass mobile resource limits [29]. For instance, the feature extraction, data storage, and matching components of a selfie biometric system can be moved to a cloud infrastructure, while leaving only the sensing task in the device. This is an example of a *biometrics-in-the-cloud* paradigm [10].

There is an increased interest in performing biometric recognition in mobile devices and as a cloud-based service [2, 4, 9, 12, 16]. In [5], a framework for cloud-based face recognition emphasizing the parallelization of recognition tasks across multiple servers is introduced. Performing biometric recognition in mobile devices and as a cloud-based service has also been adopted widely in the biometric recognition industry. For example, Zoloz provides cloud-based selfie biometric authentication solutions, which are used by around 50 banks and 200 million users in Asia. FacePhi offers a cloud-based mobile facial recognition solution, Selphi, that enables mobile banking users to access their accounts just by taking a selfie.

There are a variety of models for providing biometrics in the cloud. *Biometrics-as-a-Service* (BaaS) is a model where the biometrics-in-the-cloud architecture is offered by a service provider [28]. If the infrastructure allows for component developers to develop and incorporate custom components in the cloud (e.g., feature extraction or matcher modules), then it is referred to as *Platform-as-a-Service* (PaaS) [18]. Broadly speaking, some PaaS providers, such as Bungee Labs and SalesForce.com, provide a framework that allows independent software vendors (ISV) to develop extensions or add-ons to the provider's core application [3]. A key contribution of this chapter is to present a similar framework for cloud-based selfie biometric authentication known as *Selfie-Biometrics-as-a-Service* (SBaaS) that allows the developers of biometric recognition algorithms to actively contribute to the SBaaS system. This is achieved by creating an interface for uploading algorithms and a scheme for selecting algorithms and rendering micropayments to the developers. Having such an infrastructure in place has the benefit of promoting innovation and reducing costs for the BaaS by allowing the development of its key components to be outsourced [32].

Authentication in the cloud may raise questions regarding the preservation of information confidentiality and the use of secure authentication methods in the cloud. Privacy and security of the biometric data can be achieved by combining cloud authentication modules with biometric security architectures involving the conversion of biometric features into secure signals that can be stored in the biometric database but are still usable for authentication.

In this chapter, we present a cloud-based framework SBaaS for performing *selfie biometric recognition* using smartphones or other mobile devices as sensors and demonstrate a reference implementation of this framework using ocular recognition as a specific example. The salient features of this framework include the following:

1. Smartphones and other mobile devices, including the cameras resident in them, require no modification from their stock hardware configuration.
2. Computationally intensive tasks such as segmentation, feature extraction, and matching are outsourced to the cloud.
3. Software developers can upload their own biometric matching algorithms to the cloud. Thus, the cloud hosts multiple matching algorithms pertaining to multiple developers.
4. Enabling developers to upload matching algorithms creates an environment where the value of algorithms is measured by their in-application performance, creating incentives for competition and innovation.
5. The matching algorithm is automatically selected based on the characteristics of the input images and the performance of the algorithm as evaluated on sequestered data.
6. Every time an algorithm is selected for matching, its developer is credited under a *micropayment* model.

In addition to the SBaaS framework, we also present a *secure* Selfie-Biometrics-as-a-Service (SSBaaS) framework, in which the biometric features are converted into secure signals using secure biometric architectures to preserve the privacy information of the user.

14.2 Architecture

This section develops a general framework for SBaaS, which can be implemented for any biometric modality. The components of the system are the *user interface*, *developer interface*, *biometric database*, and cloud-based *computing infrastructure*. The user interface is an application, which may be embodied as a mobile-enabled web application or a native smartphone app, through which users submit matching requests. In particular, users use the interface to submit a selfie image to be compared with a corresponding enrollment selfie image stored in the biometric database, where the biometric database is a storage location for enrollment selfies. The developer interface is a virtual machine having identical software as the computing infrastructure for developing and submitting matching algorithms to the system. The computing infrastructure consists of a cloud server for executing matching requests using developer-submitted algorithms. The architecture is depicted in Fig. 14.1.

14.2.1 Cloud Computing Characteristics

The definition of cloud computing [19] encompasses several elements that the SBaaS architecture provides. Users can submit matching requests through a web interface,

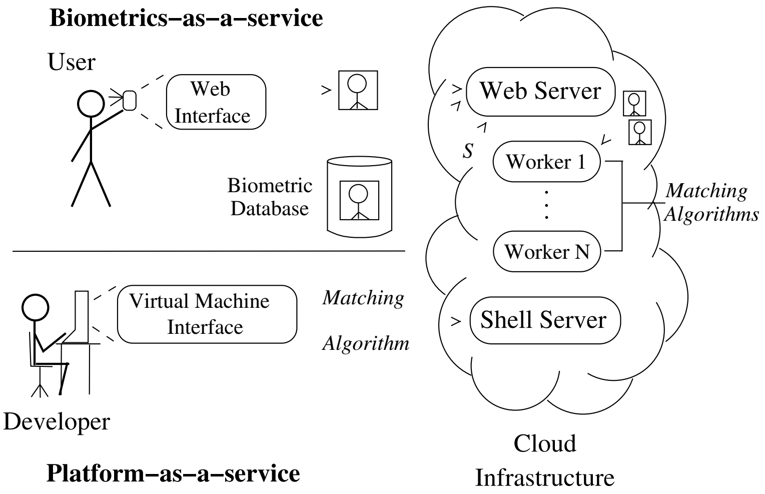


Fig. 14.1 Proposed SBaaS architecture. A user submits a matching request by uploading a probe selfie captured using the camera on a mobile device. The probe selfie from the user and the corresponding enrollment selfie from the biometric database are submitted to the cloud infrastructure. The comparison is performed by a worker process. Matching algorithm developers use the virtual machine interface to develop and submit their algorithms to the cloud infrastructure over a shell session for deployment to the worker processes

and the requests are automatically processed by available servers, providing *on-demand self-service*. The web interface is designed for both mobile and desktop use, incorporating *broad network access*. *Resource pooling* is implemented such that multiple matching requests are distributed among servers, automatically balancing the request load as needed.

Additional servers can be added to the system rapidly, as the operating system installation and software provisioning are fully automated, providing *rapid elasticity*. The computing framework software is designed to execute matching requests across any number of servers. The execution of each matching request is tracked, and users are charged for service in proportion to the number of completed requests and according to the selected algorithms (i.e., not all algorithms will be priced the same). Matching algorithm developers are credited for every matching request that uses their algorithm, making this an instance of *measured service*.

Cloud services are classified according to the level of abstraction at which the users and developers interact with the infrastructure. In the SBaaS architecture, users perform matching operations by submitting requests through a web interface. In the context of biometrics, this architecture is an instance of *biometrics-as-a-service*, a model for providing biometric recognition functionality through a service provider [28]. A virtual machine containing an operating system and pre-installed software that is identical to the cloud infrastructure is provided to the software developers, making this an instance of *Platform-as-a-Service*. The use of a virtual machine ensures that developed algorithms are binary compatible with the infrastructure and obviates the need for developers to provision their own development environments.

14.2.2 User Interface

The *user interface* shown in Fig. 14.1 is a web application that is accessible on both mobile and desktop browsers. This interface is used by the user to submit a probe selfie for matching. The probe selfie and the enrollment selfie from the biometric database are sent to the cloud infrastructure where preprocessing is done to select the most suitable matching algorithm. The matching algorithm can be selected automatically by the system, or the user can also select the matching algorithm through the user interface. The selected algorithm is executed on the selfies and a match score is returned. Even though Fig. 14.1 shows that the user interface is embodied as a web application, it could also be implemented as a native app that runs directly on the smartphone. While it could be a stand-alone app, it could also be integrated into another app, such as a banking or e-commerce app.

14.2.3 Biometric Database

The *biometric database* stores the enrollment selfies of the users. When a user submits a matching request through the user interface by uploading their selfie to the cloud, the corresponding enrollment selfie from the biometric database is also uploaded to the cloud infrastructure for authentication. The biometric database may be centralized, or it may be distributed. For instance, the biometric database could be stored on the user's smartphone as part of the image gallery or it could also be a part of the cloud infrastructure. Another example of a biometric database is that the user can store their enrollment selfie in their personal accounts with a storage provider such as Dropbox, which would cater to the needs of a nomadic user for access from multiple devices. Alternatively, the biometric database could be held by an entity such as a bank or an e-commerce site, or it could also be hosted by a third-party authentication service provider. This idea of third-party authentication service provider is analogous to the password authentication service rendered by, for example, Facebook or Google.

14.2.4 Computing Infrastructure

The computing infrastructure executes matching requests using developer-submitted algorithms. The user submits a matching request to the web server by uploading a probe selfie and, optionally, selecting a matching algorithm. The other selfie for the matching request is the enrollment selfie from the biometric database. If the user does not select an algorithm, the system selects one automatically. The server stores the selfies as data files in the user's *data directory* and creates a *job file* in the user's *job input queue* containing parameters required for matching, such as filesystem paths to the input files and to the matching algorithm, if an algorithm has been selected.

The *job manager* preprocesses the job file and moves it to the user's *job running queue*. Preprocessing is performed as follows. If the user selected a specific matching algorithm, no action is taken during preprocessing. If no algorithm was selected by the user, the system selects an algorithm automatically depending on the characteristics of the probe selfie. A table of matching algorithms and the number of times each has been executed is updated by the job manager, incrementing the number of executions for the selected algorithm. Following preprocessing, the job manager creates a *task file* containing the paths to the input files and the selected matching algorithm and places the task file in the *user task input queue*.

The *task controller* determines the number of tasks contained in all users' task input queues and schedules tasks for execution such that all users receive an equal share of the available processing cores, which is an instance of *fair scheduling*. The system is implemented such that other scheduling policies may be incorporated. To schedule an individual task for execution, the task controller moves the task file from the user's task input queue to the *global task input queue*.

A *generic worker process* running on one of the worker nodes reads a task in the global task input queue and moves it to the *global task running queue*. The generic worker process executes the matching operation, saves the result in the task file, and moves it to the *global task output queue*. The task controller moves the task from the global task output queue to the *user task output queue*. The task file is read by the job manager, and the matching result is stored in the job file and moved to the *job output queue*. The web interface reads the matching result from the job file and displays it to the user.

14.2.5 Developer Interface

The *developer interface* is a *virtual machine* (VM) that contains software for developing algorithms for submission to the cloud infrastructure. The operating system and software environment on the VM are configured identically to the environment on the computing infrastructure (i.e., the cloud server). Identical configuration obviates the need for the algorithm developer to invest time installing and configuring a compatible development environment. The VM enables the developer to implement matching algorithms that are binary compatible with the infrastructure, and upload scripts and executables directly for use.

The VM is distributed over the Internet as an archive containing a disk image of the preinstalled and configured operating system. The *software hypervisor*, which executes the virtual machine, is chosen for compatibility with as many widely used host operating systems as possible. A desktop environment having minimal resource requirements is chosen for the VM to enhance user interface performance in a virtualized environment.

The VM contains software for algorithm development, such as compilers, text editors, debuggers, and source control clients, as well as libraries commonly used for image processing applications and research. The developer must implement their algorithm such that it can be executed on a command line—a broad and general requirement that is straightforward to satisfy.

14.3 A Reference Infrastructure Implementation

This section describes a reference implementation of the general SBaaS architecture described in Sect. 14.2. The user interface implementation is first described, followed by the developer interface. Finally, details of the computing infrastructure implementation are given. For the implementation given in this section, the biometric database is considered to be the image gallery on the user's smartphone or tablet.

14.3.1 User Interface

The user interface is implemented using Mobile-Google Web Toolkit (MGWT)[1] which is a software framework for developing mobile web applications. MGWT is an extension of Google Web Toolkit (GWT),[2] which is a Java-based framework, for creating efficient and optimized browser-based applications. GWT is an open-source completely free framework that helps developers to build high-performance web applications without having expert skills in JavaScripting or browser quirks. Google also uses GWT in many of its products such as Inbox, Calendar, Adwords, and AdSense.

While GWT can help build fast desktop applications using Java, it lacks widgets and animations for developing mobile apps. MGWT closes this gap—MGWT provides mobile widgets, smooth animations, touch support, and much more. One can use MGWT to build highly optimized Java-based AJAX applications that are compatible with all browsers, including Android and the iPhone mobile browsers. We used MGWT 1.1.2 along with GWT 2.7 and Eclipse to develop the user interface. A few other Java-based API's were also used along with MGWT to develop the functionality of the user interface.

The steps taken by a user to submit a job through the web interface and obtain the results of the matching are as follows:

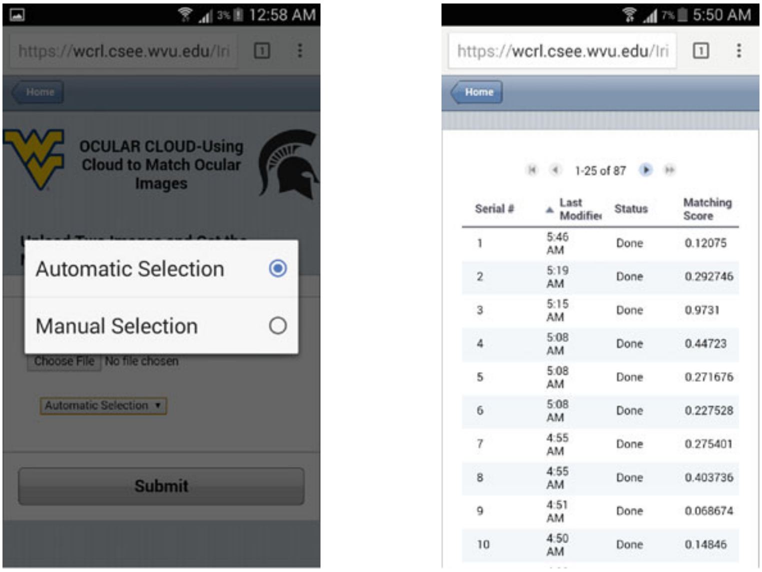
1. The web application is accessed by pointing a mobile browser to a known Web site.
2. After logging into the application, the user can either view their previous job submissions or submit a new job.
3. If the user wishes to submit a new job, they can upload a probe selfie, a gallery selfie, and either explicitly select a matching algorithm or allow the interface to select an appropriate algorithm based on image characteristics, as shown in Fig. 14.2a.
4. Upon submitting the job request, the user will be redirected to the Job History page. Shown in Fig. 14.2b is the Job History page view. This page provides details about all previous jobs submitted by the user. The user can view the complete details of a particular job—including the input images and the matching score—by clicking on the associated job.

14.3.2 Computing Infrastructure

The computing infrastructure used in this reference implementation is a heterogeneous cloud of servers, each having a varying number of processing cores and main

[1] http://www.m-gwt.com/.

[2] http://www.gwtproject.org/.


12:58 AM
https://wcrl.csee.wvu.edu/Iri
Home
OCULAR CLOUD-Using Cloud to Match Ocular Images
Automatic Selection ●
Manual Selection ○
Choose File No file chosen
Automatic Selection ▾
Submit
5:50 AM
https://wcrl.csee.wvu.edu/Iri
Home
1-25 of 87

Serial #	Last Modifie	Status	Matching Score
1	5:46 AM	Done	0.12075
2	5:19 AM	Done	0.292746
3	5:15 AM	Done	0.9731
4	5:08 AM	Done	0.44723
5	5:08 AM	Done	0.271676
6	5:08 AM	Done	0.227528
7	4:55 AM	Done	0.275401
8	4:55 AM	Done	0.403736
9	4:51 AM	Done	0.068674
10	4:50 AM	Done	0.14846

(a) Algorithm Selection page

(b) Job History page

Fig. 14.2 Screen shots of the mobile web app

memory. A single server acts as a router between the Internet and remaining servers and hosts the web and shell servers. This server is denoted as the *head node*. The remaining servers execute matching requests and are denoted as *worker nodes*. The operating system on all nodes is *Ubuntu Linux*. All data are stored in the head node and shared to the worker nodes using the standardized distributed file system protocol *Network File System* (NFS). The software components implementing the job manager, task controller, and generic worker are designed to work independently of one another, communicating through files on the file system. This architecture allows the components to be reused with little or no modification to the code, consistent with the UNIX philosophy and the notion of a *microservice* [35].

The job manager is implemented as a MATLAB® program that runs persistently on the head node within a *GNU screen* session. When a user submits a matching request using the web interface, the interface creates a data file (in MATLAB's *.mat* format) containing (a) the paths to the input images and (b) the user's algorithm selection option; this file is stored in the user's home directory. The job manager creates a task file—also in *.mat* format—containing paths to the images and the algorithm to execute.

Like the job manager, the task controller is implemented as a MATLAB program on the head node that is run in a GNU screen session. The task controller schedules a user's task for execution when the worker node resources become available. Exactly one matching request may be executed for every processing core available on the worker nodes. Once this limit is reached, further matching requests must wait until a core becomes available.

A matching request is executed by a generic worker process running on a worker node. The generic worker process is a MATLAB program which executes the algorithm specified in the task file. The task file specifies an *entry function* for the matching algorithm, which is a MATLAB function implemented by the algorithm developer to initiate algorithm execution. Since the algorithm developer has full control of the entry point function, they may execute a program implemented in any language which can be executed on the Linux command line by using the MATLAB feature to execute shell commands.

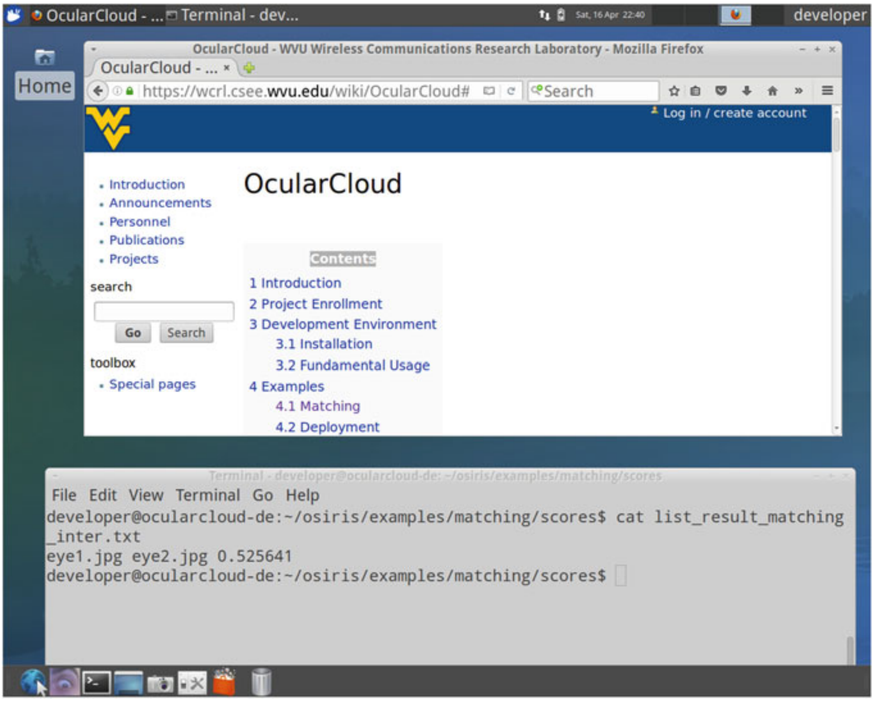
Once the algorithm execution is complete, control is returned to the generic worker, which stores matching results in the task file. The task file is consumed by the job manager, which stores the matching result in the job file. The job file is passed through the queues to the web interface, which displays the matching result to the user. In this reference implementation, the matching score is sent back to the user.

14.3.3 Developer Interface

The developer interface is implemented as a virtual machine using *Ubuntu* as the operating system. This is the same operating system installed on the cloud infrastructure nodes, which simplifies the deployment of matching algorithms. The software tools and libraries used for compiling algorithms in the developer interface exactly match those on the infrastructure, enabling the developer to deploy binaries directly. Virtualbox was chosen as the hypervisor as it is freely available for all major computing platforms (Windows, OSX, and Linux). An example of the developer interface is shown in Fig. 14.3.

The virtual machine is distributed through a publicly accessible Web site as a compressed archive, which expands to a single Virtualbox Disk Image (VDI) file. The developer specifies the VDI file as the disk image for a virtual machine in Virtualbox, and the developer interface is immediately available. Downloading and executing a virtual machine image is much simpler than a conventional provisioning process where the developer personally installs Ubuntu and the required software.

The virtual machine contains various machine learning frameworks, an open-source computer vision library Open CV *2.4.11* [7], and standard utilities for software development in the Linux environment such as GNU Emacs and the GNU compiler collection. Documentation for using the interface is provided as a wiki, which is linked via the interface desktop. The developer deploys their algorithm by uploading the required scripts, executables, and data files to their home directory in the cloud and submitting a request for integration to the infrastructure administrator.



OcularCloud - ...
Terminal - dev...
Sat, 16 Apr 22:40
developer
Home
OcularCloud - WVU Wireless Communications Research Laboratory - Mozilla Firefox
OcularCloud - ...
https://wcrl.csee.wvu.edu/wiki/OcularCloud#
Search
Log in / create account
Introduction
Announcements
Personnel
Publications
Projects
search
Go
Search
toolbox
Special pages
Terminal - developer@ocularcloud-de: ~/osiris/examples/matching/scores
File Edit View Terminal Go Help

OcularCloud

Contents

1 Introduction
2 Project Enrollment
3 Development Environment
3.1 Installation
3.2 Fundamental Usage
4 Examples
4.1 Matching
4.2 Deployment

```
developer@ocularcloud-de:~/osiris/examples/matching/scores$ cat list_result_matching
_inter.txt
eye1.jpg eye2.jpg 0.525641
developer@ocularcloud-de:~/osiris/examples/matching/scores$
```

Fig. 14.3 Virtual machine implementing the developer interface. The web browser displays the developer documentation wiki. A terminal window shows the match score for two images as computed by a matching algorithm

14.4 An Operational Example

Ocular biometrics is the combination of multiple modalities in and near the eye region, such as the iris and periocular region [1, 25, 36]. In this operational example, we focus on the iris and periocular modalities as examples (together referred to as “ocular”). We performed two sets of experiments that illustrate the potential benefits of the framework. The first experiment shows that different algorithms provide different matching performance, thereby motivating the need for a system that supports a plurality of algorithms. The second experiment evaluates the performance of the system when the algorithm is automatically selected.

14.4.1 An Illustration of Algorithmic Diversity

The dataset used for the first experiment is the ND-IRIS-0405 iris dataset [6]. This dataset contains 64,980 images corresponding to 356 unique subjects and 712 unique irises. For our evaluation, we use iris images of 12 subjects and 12 images of the same iris per subject. In total, we used 144 images. All of the matching algorithms used by the example system are based on Open Source for IRIS (OSIRIS), which is a well-known open-source iris recognition system developed in the framework of the BioSecure project.[3] Specifically, OSIRIS *4.1* was used,[4] which is composed of four processing modules—segmentation, normalization, encoding, and matching. Gabor filters are applied to the normalized iris image, and the resulting phasor responses are quantized into a binary feature set. The Hamming distance measure is used to compare the binary feature sets of two iris images in order to obtain the final matching score.

In order to mimic the use of multiple algorithms, different Gabor filter parameters were selected for the OSIRIS algorithm, resulting in different sets of Gabor filters. This was accomplished by changing the *sizes* of the Gabor filters, or by changing the *number* of Gabor filters. Gabor filter coefficient sizes are defined in terms of the coefficient matrix ($m \times n$). We used five different Gabor filter parameter sets **A, B, C, D**, and **E** for this experiment. The **A, B**, and **C** parameter sets have 2 Gabor filters each, with coefficient matrix sizes of 9×15, 9×27, and 9×51, respectively. Parameter sets **D**, and **E** have 4 and 6 Gabor filters each, respectively. When used with OSIRIS, each parameter set is viewed as a different algorithm, which we denote Algorithm **A**, Algorithm **B**, Algorithm **C**, Algorithm **D**, and Algorithm **E**.

The following experiment was performed to evaluate and compare the performance of the three algorithms **A**, **B**, and **C**. False accept rate (FAR), false reject rate (FRR), and genuine accept rate (GAR) are computed for the test dataset of 144 images. Based on the number of subjects ($N = 12$) and the number of images ($t = 12$) per subject, we obtain $Nt(t-1)/2 = 792$ genuine scores and $(N(N-1)t^2)/2 = 9504$ imposter scores. The ROC (GAR vs FAR) curve at various threshold points for the first experiment is shown in Fig. 14.4. Algorithm **B** with 2 Gabor filters of size 9×27 performs marginally better than the other two algorithms. But it can be observed from the curves that there is no clear winner. However, these curves suggest that different algorithms may be needed depending upon operational requirements of FAR and/or GAR.

A similar experiment was performed for comparing the algorithms (**A, D**, and **E** which have a different number of filters (2, 4, and 6, respectively). The ROC curve for this experiment is shown in Fig. 14.5. This figure quite evidently solidifies the assumption and the motivation behind the solution proposed as we can clearly see that one algorithm never comes out on top as the curves do intersect at a number of points. So, again as already stated depending on the images, thresholds, and region of operation, a different algorithm can be selected for matching of the selfie ocular images.

[3] http://biosecure.wp.tem-tsp.eu/.

[4] http://svnext.it-sudparis.eu/svnview2-eph/ref_syst//Iris_Osiris_v4.1.

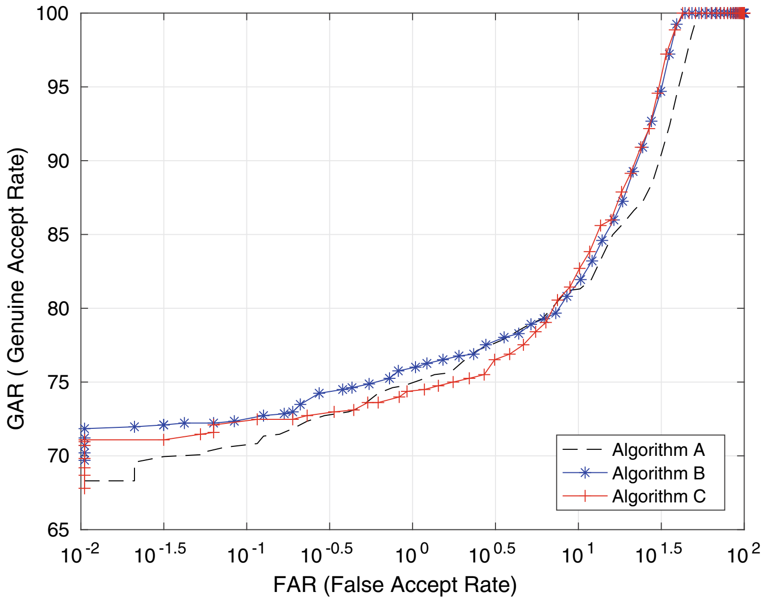


Fig. 14.4 ROC curves for three algorithms that each use two filters, but with different sizes

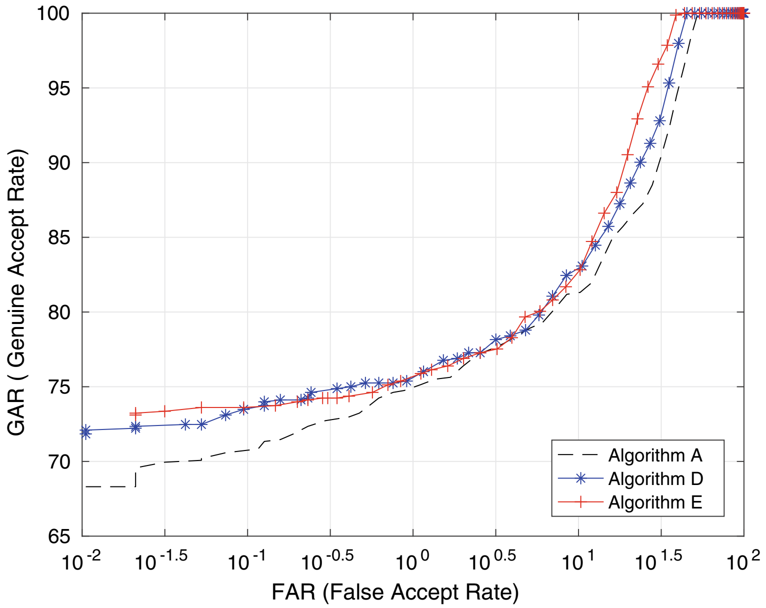


Fig. 14.5 ROC curves for three algorithms that each use a different number of Gabor filters

14.4.2 An Illustration of Automatic Algorithm Selection

In order to evaluate our SBaaS framework reference implementation, we conducted an experiment by using the web app on a smartphone. The trials were conducted by using a selfie image of the ocular region as one input and an ocular image from the phone gallery as the second input. Each trial entailed matching two images, and the experiment entailed 320 such trials. The trials consisted of both genuine and impostor image pairings.

Once the selfie images were uploaded to the cloud, the algorithm to be executed on the input images was selected automatically. Based on the input image characteristics, a particular algorithm was automatically invoked by the system at the time of authentication. In this experiment, three algorithms were used. The first was an OSIRIS-based iris recognition algorithm (“OSIRIS”), the second was a custom periocular matching algorithm (“Periocular”), and the third was a neural network iris matcher (“NN”). The selection method first computes the radius of the iris region in the input ocular image and uses this to select one of the three algorithms. In particular, the range between minimum and maximum radius of the iris is divided into three parts using two thresholds. If the radius of the iris is below the first threshold, the algorithm “Periocular” is selected. If the radius of the iris is between the first and the second threshold, then “OSIRIS” algorithm is selected for execution. Finally, if the radius of the iris is above the second threshold, then “NN” algorithm is executed.

Table 14.1 gives the number of times each algorithm was executed during the 320 trials conducted in the experiment. Besides hosting three completely different algorithms, it is possible for the cloud to host several instances of the same algorithm, where the different instances use different parameters. The experiment conveys the main theme of the proposed framework, i.e., depending on the input images, a different algorithm is selected each time and the developer for that selected algorithm is rendered a micropayment. This experiment shows that the proposed framework is feasible and creates an innovative and competitive ecosystem that benefits both software developers and end users.

For the above experiments, the radius of the iris has been considered as one of the variables to be used for automatic selection of the algorithm. However, there are a lot of other variables that can be used to automate the selection of the algorithm to be executed on the pair of input selfies. An example of other variable to be used for

Table 14.1 Table showing the number of executions for each algorithm for a total of 320 trials

Serial No.	Developer name	Algorithm name	Modality	Number of executions	Number of executions in %
1	TMPS	OSIRIS	Iris	102	31.9
2	ROSS	Periocular	Ocular	68	21.2
3	VT	NN	Iris	154	46.9

automation could be the matching score generated by the algorithm. An algorithm that gives the best matching score can be selected and the micropayment can be rendered to the corresponding developer. However, this entails running all the algorithms in the cloud on a given pair of inputs which would make the system really slow as it would lead to a huge computational cost.

Another example to automate the selection could be dependent on the micropayments that are being rendered. Assuming that the micropayment amount for an algorithm is decided by the developer uploading their algorithm (it could be the licensing fee for using the algorithm). In that case, the algorithm selection could be automated based on the micropayment being rendered. Less micropayment algorithm could be selected more number of times.

Another way could be to select the strategy for automatic selection of the algorithm dynamically depending on the load of the system. If there is a heavy load and the number of selfie images to be matched in the queue is above a threshold, then the algorithm selection could be switched from the original automation strategy to a new strategy. The new strategy could be dependent on the computational cost of each algorithm, and this could be used to generate faster matching scores and reduce the load.

14.5 Secure Selfie-Biometrics-as-a-Service

In the general SBaaS framework discussed in Sec. 14.2, the enrollment selfie is stored in the biometric database. As in the reference implementation, the biometric database could simply be the image gallery on the user's device. Alternatively, the biometric database could be stored in the cloud or at a private server hosted by a bank, an e-commerce site, or a third-party authentication service provider. In these alternate scenarios, the security of the stored biometric data is critical and is not sufficiently integrated into the previously discussed SBaaS framework. In this section, we present an improved model that provides a solution for the security of the stored biometric data. This new model is termed "Secure Selfie-Biometrics-as-a-Service (SSBaaS)." A general SSBaaS architecture is given in Fig. 14.6. The SSBaaS architecture consists of two modules: a *feature extraction module (FEM)* and a *biometric security module (BSM)*. The FEM consists of feature extraction algorithms uploaded by the developers, and the BSM consists of the common back-end components required to keep the biometric secure in the SSBaaS architecture.

14.5.1 Feature Extraction Module

The SSBaaS model differs from the previously discussed SBaaS model in terms of how the developer contributes to algorithms. In the SBaaS model, the developer uploads an end-to-end biometric matching algorithm to the cloud. However, in the

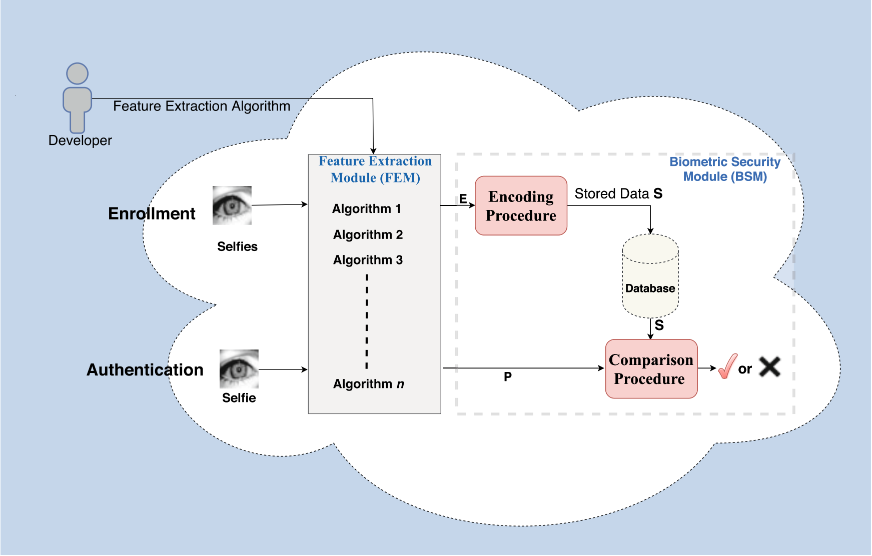


Fig. 14.6 General secure selfie-biometrics-as-a-service architecture

SSBaaS model, the developer only uploads algorithms for the extraction of features from the enrollment and probe selfies. The FEM contains all the feature extraction algorithms uploaded by the developers. The features extracted from each selfie have to be in the form of a binary vector. To preserve the privacy of the user, the binary feature vector is not directly stored in the database. Instead, the binary feature vector is passed through the BSM to generate a secure biometric template that is stored in the database. It is important to articulate to the developers the requirements that are enforced on the biometric feature extraction algorithms to generate the biometric feature vectors that are compatible for use in the BSM.

The statistical and privacy-preserving properties desired for the biometric feature vectors are as follows:

1. A bit in the feature vector is equally likely to be zero or one, which helps in maximizing the entropy of the feature vector.
2. A given bit in a feature vector provides no information about any other bit in the feature vector, which implies different bits in the feature vector are independent of each other.
3. The feature vector of one person provides no information about the feature vector of the other person, which implies inter-user independence.
4. Strong intra-user dependence, which implies different measurements of the same user are related by a binary symmetric channel (BSC) with crossover probability p where p is much smaller than 0.5.

Upon registering with the system, a developer is required to be bound to a set of constraints that enforce the above properties. Satisfying the above properties not only ensures good matching performance but also provides nice privacy-preserving features. This also benefits the developers by improving the chances of their algorithm being selected and micropayment being rendered to their accounts. However, designing feature vectors to possess such privacy-preserving properties forces a compromise between robustness and discriminability of the feature values, which in turn affects the accuracy (FRR and FAR) of the system, underscoring the fact that privacy at this time comes at the price of performance.

14.5.2 Biometric Security Module

In the SSBaaS model, it is assumed that the enrollment biometric information is stored in the cloud or on a private server. Consequently, the SSBaas model must preserve the biometric information confidentiality of the user. The leakage of biometric information stored in the cloud to an adversary constitutes a serious threat to security and privacy because if an adversary gains access to a biometric template, he can potentially obtain the stored user information. The attacker can use this information to gain unauthorized access to the system by reverse engineering the system and creating a physical spoof. Furthermore, an attacker can abuse the biometric information for unintended purposes and violate user privacy [21].

To alleviate such security and privacy concerns, secure biometric schemes have been developed to allow for authentication without requiring the enrollment biometric template to be stored in its raw format. BSM, which is shown as a part of Fig. 14.6, presents a general secure biometric scheme. The functionality of BSM is to develop a suitable *encoding procedure* for transforming enrollment biometric data into a template to be stored in the cloud and also to develop a *comparison procedure* for matching the probe biometric data with the stored template to produce an authentication decision. The BSM constitutes the back end of the SSBaaS architecture and is common to all developers and users of the complete system. All the feature extraction developers leverage the same BSM and have no direct access to the BSM. Rather, they only contribute to the FEM.

There are four main specific implementations of secure biometric schemes that are widely used: *fuzzy commitment, secure sketch, secure multiparty computation*, and *cancelable biometrics* [26]. *Fuzzy commitment* and *secure sketch* are biometric cryptosystem methods and are usually implemented with error-correcting codes and provide information-theoretic guarantees of security and privacy (e.g., [14, 15, 20, 23, 30]). *Secure multiparty computation* architectures are distance based and use cryptographic tools. *Cancelable biometrics*, which is a transformation based method, uses revocable and non-invertible user-specific transformations for distorting the enrollment biometric (e.g., [17, 27, 34, 37]), with the matching typically performed in the transformed domain. Fuzzy commitment, secure sketch, and cancelable biometrics architectures are described briefly below, treating each as a special manifestation of the BSM in 14.6.

Fuzzy commitment, a classical method of biometric protection, was first proposed by Juels and Wattenberg [15] in 1999. Fuzzy commitment is a key-binding method of biocryptosystem, and the encoding procedure involves combining a randomly generated vector $\mathbf{Z}$ with the enrollment biometric feature $\mathbf{E}$ resulting in the stored data $\mathbf{S}$. The comparison procedure checks whether the randomly generated vector $\mathbf{Z}$ is exactly recovered using the probe feature vector $\mathbf{P}$ and $\mathbf{S}$. There are many methods of implementing this fuzzy commitment scheme. However, a common method is to use error control coding (ECC). An example of using ECC for fuzzy commitment involves constructing the stored data as $\mathbf{S} = \mathbf{G}^T\mathbf{Z} \oplus \mathbf{E}$, where $\mathbf{G}$ is the generator matrix of an ECC. During authentication, the probe feature vector $\mathbf{P}$ is combined with $\mathbf{S}$ using $\mathbf{S} \oplus \mathbf{P}$. Next, using ECC decoding, the system attempts to decode the random message $\mathbf{Z}$ and allows access only if it is successful.

Secure sketch is a key generation method where some helper data or a sketch $\mathbf{S}$ is derived from the enrolled biometric feature vector $\mathbf{E}$ and stored in the database. The probe is given access when the probe biometric feature vector $\mathbf{P}$ is consistent with the stored secure sketch $\mathbf{S}$. The sketch $\mathbf{S}$ should be constructed so that it reveals little or no information about $\mathbf{E}$. Similar to fuzzy commitment, a common method of implementing secure sketch is to use ECC. In this method, ECC is applied to the biometrics or the feature vector to generate a sketch, which is stored in the database. The secure sketch $\mathbf{S}$ is constructed as $\mathbf{S} = \mathbf{HE}$; which is constructed as a syndrome of an ECC with parity check matrix $\mathbf{H}$. A legitimate probe biometric $\mathbf{P} = \mathbf{E}$ would be a slightly error-prone version of $\mathbf{E}$. Consequently, authentication can be accomplished by attempting to decode $\mathbf{E}$ given $\mathbf{P}$ and $\mathbf{S}$.

Cancelable biometrics involves transforming or distorting the enrollment biometric with a non-invertible user-specific transformation. The transformation in cancelable biometrics is a one-way transformation and can be applied either to the original biometric or in the feature domain. The advantage of using one-way transformations is that they are non-invertible and therefore the original biometric cannot be recovered easily. This transformation is revocable as well, which means that if the biometric is compromised, a new transformation can be applied to generate the cancelable template. This helps in protecting the privacy and also deters cross-matching since a different transformation can be used for a different application. Cancelable biometrics was first proposed by Ratha et al. [27], following which, there have been various different methods of generating cancelable biometric templates. Some of the popular methods use non-invertible transforms [27], bio-hashing [17], salting [37], and random projections [34]. Literature surveys on cancelable biometrics can be found in [26] and [24].

The secure biometric architectures explained above could be extended to include multiple biometric traits of a user [8, 21, 22, 31, 33]. Nagar et al. [21] developed a multimodal cryptosystem based on feature-level fusion using two different security architectures, fuzzy commitment, and fuzzy vault. In [31], face and fingerprint templates are concatenated to form a single binary string, and this concatenated string is used as input to a secure sketch scheme. In [33], a feature-level fusion framework is presented to generate a shared representation from each user's multiple biometrics. For each user, a selection of a different set of reliable and discriminative features from the shared representation is performed to generate a cancelable biometric template. This cancelable template is passed through an appropriate error-correcting decoder to find the closest codeword, which is hashed to generate the final secure multimodal template.

14.5.3 Reference Implementation of SSBaaS

A reference implementation of the SSBaaS architecture is shown in Fig. 14.7. In the reference implementation, the focus is on presenting a specific manifestation of the BSM. During enrollment, the user submits a selfie (i.e., enrollment selfie), which is transmitted to the cloud. In the cloud, depending on the learning or the state of the system, one or more feature extraction algorithms from the FEM are executed for the enrollment selfie. Initially, when the system has not learnt anything, all the feature extraction algorithms may be executed by the system for a given enrollment selfie. However, with an increase in enrollments, the system learns the best feature extraction algorithm for a given enrollment selfie, depending on certain learning criterion. Examples of these learning criteria are discussed later at the end of this section. For now, we can assume each feature extraction algorithm from the FEM is executed and one enrollment feature vector, denoted by E, is generated for each algorithm. Consequently, the number of enrollment feature vectors for each enrollment selfie is equal to the number of feature extraction algorithms (say n) in the FEM. For clarity of exposition, only one feature vector **E** is shown at the output of FEM in Fig. 14.7.

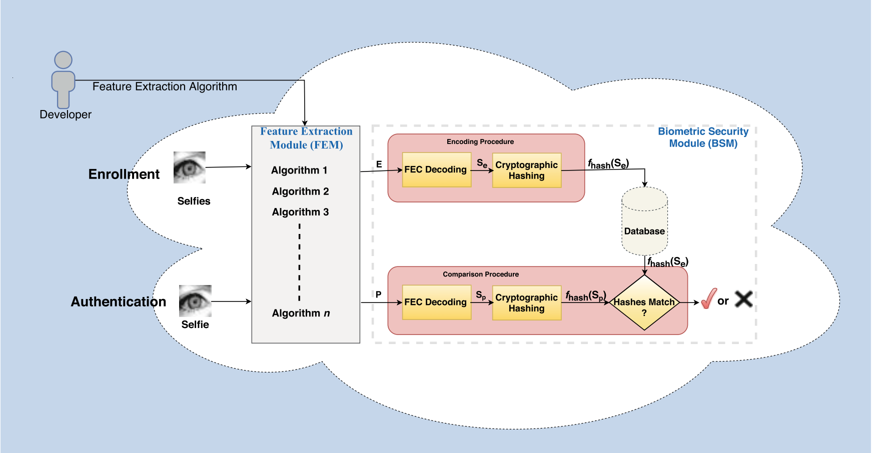


Fig. 14.7 Reference implementation of SSBaaS

The enrollment binary feature vector **E** is now passed to the BSM module for further processing and generation of secure biometric template in the database. The encoding procedure in this BSM consists of two steps: *forward error correction (FEC) decoding* and *cryptographic hashing*. The FEC decoding in this implementation is the equivalent of a secure sketch template protection scheme. In a secure sketch scheme, sketch or helper data are generated from the user's biometrics, and this sketch is stored in the access control database. There are many methods of implementing this secure sketch scheme. However, a common method is to use error control coding. In this method, error control coding is applied to the biometrics or the feature vector to generate a sketch which is stored in the database. Similarly, in this implementation, the FEC decoding is considered to be the error control coding part required to generate the secure sketch. The enrollment binary feature vector **E** generated from the FEM is considered to be the noisy codeword of some error-correcting code. This noisy codeword is decoded using FEC decoding, and the output of the decoding is the biometric secure sketch $\mathbf{S_e}$ that corresponds to the codeword closest to the enrollment feature vector. This biometric sketch $\mathbf{S_e}$ is cryptographically hashed to generate the secure biometric template $f_{\mathsf{hash}}(\mathbf{S_e})$, which is stored in the database. The same procedure of FEC decoding and cryptographic hashing is applied for all the n feature vectors generated by the n feature extraction algorithms for an enrollment selfie. This would imply that for each user's enrollment selfie, there would n cryptographic hashes stored in the database.

During authentication, the same process is performed. The user submits a probe selfie, which is transmitted to the cloud for authentication. A probe feature vector **P** is generated using the feature extraction algorithm in FEM. Next, the probe feature vector **P** is passed through an FEC decoder for the same error-correcting code used during the enrollment. The output of the FEC decoder is the probe biometric sketch $\mathbf{S_p}$, which is cryptographically hashed and access is granted only if this hash matches the enrolled hash. During authentication, if it is a genuine probe, the enrollment **E** and the probe vector **P** would usually decode to the same codeword in which case the hashes would match and access would be granted. Generally, if it is a legitimate probe, access would be granted. However, an adversary may use synthesized biometrics to fool and gain access to the system. Therefore, any analysis of the SSBaaS model must take into account not only authentication accuracy but also the information leakage and the possibility of attacking the system when the stored template is compromised.

Initially, when the system is still trying to determine the best feature extraction algorithm for a given user, the method discussed above could be one way of doing the enrollment, where the number of hashes stored per user is equal to the number of feature extraction algorithms in the FEM. Over a period of time, the system learns the best feature extraction algorithm for a given user depending on a number of variables. One of the variables is the execution time of the feature extraction algorithm. Some of the feature extraction algorithms may execute faster than the other algorithms. However, the execution time could be dependent on the resolution of the image, which in turn could be dependent on the device being used to capture the selfie. A *device table* could be stored in the cloud providing information as to which feature extraction algorithm works better with images from a particular device. For example, if the device is an iPhone 8, then Algorithm 3 may give fast and accurate results. In this case, iPhone 8 could be indexed with Algorithm 3. Using this table, the system can decide which algorithm needs to be used if the user operates a particular device; thus, it should be able to generate the enrollment feature vector only using the corresponding algorithms from the device table. However, this is just one method of deciding the best feature extraction algorithm. There could be other variables such as matching accuracy and micropayment cost that could be used to decide the best feature extraction algorithm for particular user enrollment. This is a design decision, and it might differ depending on the system requirements.

14.6 Summary

In this chapter, we presented a Selfie-Biometrics-as-a-Service framework for performing selfie biometric matching in a cloud environment using the sensors available in ordinary smartphones. The proposed biometrics-as-a-service paradigm enables users to perform biometric matching in a web interface. Moreover, the Platform-as-a-Service model enables the developers of recognition technology to upload their algorithms to the cloud. By selecting algorithms for execution and rendering micropayments to the corresponding developer, continuous algorithm innovation is encouraged. A reference implementation and an operational example have been presented demonstrating that the architecture is feasible in the form of a case study based on ocular recognition. Additionally, an overview of a secure Selfie-Biometrics-as-a-Service model has been discussed with a major focus on biometric template security in the cloud.

Acknowledgements This research was funded by the Center for Identification Technology Research (CITeR), a National Science Foundation (NSF) Industry/University Cooperative Research Center (I/UCRC).

References

1. Alonso-Fernandez F, Bigun J (2015) Near-infrared and visible-light periocular recognition with gabor features using frequency-adaptive automatic eye detection. IET Biom 4(2):74–89
2. Barra S et al (2015) Ubiquitous iris recognition by means of mobile devices. Pattern Recognit Lett 57:66–73
3. Beimborn D, Miletzki T, Wenzel S (2011) Platform as a service (PaaS). Bus & Inf Syst Eng 3(6)
4. Bharadi VA, D'silva GM (2015) Online signature recognition using software as a service (SAAS) model on public cloud. In: International conference on computer, communication and automated, pp 65–72
5. Bommagani AS, Valenti MC, Ross A (2014) A framework for secure cloud-empowered mobile biometrics. In: Proceeding of IEEE military communications conference, pp 255–261
6. Bowyer KW, Flynn PJ (2010) The ND-IRIS-0405 iris image dataset. University of Notre Dame, CVRL

7. Bradski G (2000) The opencv library. Dr. Dobb's J Softw Tools Prof Program 25(11):120–123 (2000)
8. Canuto AM, Pintro F, Xavier-Junior JC (2013) Investigating fusion approaches in multi-biometric cancellable recognition. Expert Syst Appl 40(6):1971–1980
9. Chow R, Jakobsson M, Masuoka R, Molina J, Niu Y, Shi E, Song Z (2010) Authentication in the clouds: a framework and its application to mobile users. In: Proceedings of the 2010 ACM workshop on cloud computing security workshop, CCSW '10. ACM, New York, NY, USA, pp 1–6. https://doi.org/10.1145/1866835.1866837
10. Das R (2013) Biometrics in the cloud. Keesing J Doc Identity, 21–23
11. de Freitas Pereira T, Marcel S (2015) Periocular biometrics in mobile environment. In: Proceeding of biometrics: theory, applications and systems (BTAS), pp 1–7
12. Jeong DS, et al (2006) Iris recognition in mobile phone based on adaptive gabor filter. In: Proceeding of international conference on biometrics (ICB), pp 457–463
13. Jillela RR, Ross A (2015) Segmenting iris images in the visible spectrum with applications in mobile biometrics. Pattern Recognit Lett 57(C):4–16
14. Juels A, Sudan M (2002) A fuzzy vault scheme. In: Proceeding IEEE international symposium on information theory, p 408. https://doi.org/10.1109/ISIT.2002.1023680
15. Juels A, Wattenberg M (1999) A fuzzy commitment scheme. In: Proceeding 6th ACM conference on computer and communications security, pp 28–36 (1999)
16. Kang JS (2010) Mobile iris recognition systems: an emerging biometric technology. Procedia Comput Sci 1(1):475–484
17. Kong A, Cheung KH, Zhang D, Kamel M, You J (2006) An analysis of biohashing and its variants. Pattern Recognit 39(7):1359–1368
18. Lawton G (2008) Developing software online with platform-as-a-service technology. Computer 41(6):13–15
19. Mell P, Grance T (2011) The NIST definition of cloud computing. In: Recommendations of the national institute of standards and technology, special publication pp 800–145
20. Nagar A, Nandakumar K, Jain AK (2008) Securing fingerprint template: fuzzy vault with minutiae descriptors. In: Proceeding 19th international conference on pattern recognition. https://doi.org/10.1109/ICPR.2008.4761459
21. Nagar A, Nandakumar K, Jain AK (2012) Multibiometric cryptosystems based on feature-level fusion. IEEE Trans Inf Forensics Secur 7(1):255–268. https://doi.org/10.1109/TIFS.2011.2166545
22. Nandakumar K, Jain AK (2008) Multibiometric template security using fuzzy vault. In: Proceeding IEEE international conference on biometrics: theory, applications and systems
23. Nandakumar K, Jain AK, Pankanti S (2007) Fingerprint-based fuzzy vault: implementation and performance. IEEE Trans Inf Forensics Secur 2(4):744–757. https://doi.org/10.1109/TIFS.2007.908165
24. Patel VM, Ratha NK, Chellappa R (2015) Cancelable biometrics: a review. IEEE Signal Process Mag 32(5):54–65. https://doi.org/10.1109/MSP.2015.2434151
25. Raghavendra R, Busch C (2016) Learning deeply coupled autoencoders for smartphone based robust periocular verification. In: 23rd international conference on image processing (ICIP). IEEE
26. Rane S, Wang Y, Draper SC, Ishwar P (2013) Secure biometrics: concepts, authentication architectures, and challenges. IEEE Signal Process Mag 30(5):51–64. https://doi.org/10.1109/MSP.2013.2261691
27. Ratha NK, Chikkerur S, Connell JH, Bolle RM (2007) Generating cancelable fingerprint templates. IEEE Trans Pattern Anal Mach Intell 29(4):561–572. https://doi.org/10.1109/TPAMI.2007.1004
28. Rose J (2016) Biometrics as a service: the next giant leap? Biom Technol Today 2016(3):7–9
29. Stojmenovic M (2012) Mobile cloud computing for biometric applications. In: 15th international conference on network-based information system, pp 654–659
30. Sutcu Y, Li Q, Memon N (2007) Protecting biometric templates with sketch: theory and practice. IEEE Trans Inf Forensics Secur 2(3):503–512

31. Sutcu Y, Li Q, Memon N (2007) Secure biometric templates from fingerprint-face features. In: Proceeding IEEE conference on computer vision and pattern recognition
32. Talreja V, Ferrett T, Valenti MC, Ross A (2018) Biometrics-as-a-service: a framework to promote innovative biometric recognition in the cloud. In: Proceeding IEEE international conference on consumer electronics (ICCE)
33. Talreja V, Valenti MC, Nasrabadi NM (2017) Multibiometric secure system based on deep learning. In: Proceeding IEEE global conference on signal and information processing, pp 298–302. https://doi.org/10.1109/GlobalSIP.2017.8308652
34. Teoh AB, Kuan YW, Lee S (2008) Cancellable biometrics and annotations on biohash. Pattern Recognit 41(6):2034–2044
35. Thönes J (2015) Microservices. IEEE Softw 32(1), 116, 113–115
36. Woodard DL, Pundlik S, Miller P, Jillela R, Ross A (2010) On the fusion of periocular and iris biometrics in non-ideal imagery. In: 20th international conference on pattern recognition (ICPR). IEEE, pp 201–204
37. Zuo J, Ratha NK, Connell JH (2008) Cancelable iris biometric. In: Proceeding IEEE international conference on pattern recognition, pp 1–4

Chapter 15
Biometric Template Protection on Smartphones Using the Manifold-Structure Preserving Feature Representation

Kiran B. Raja, R. Raghavendra, Martin Stokkenes and Christoph Busch

Abstract Smartphone-based biometrics authentication has been increasingly used for many popular everyday applications such as e-banking and secure access control to personal services. The use of biometric data on smartphones introduces the need for capturing and storage of biometric data such as face images. Unlike the traditional passwords used for many services, biometric data once compromised cannot be replaced. Therefore, the biometric data not only should not be stored as a raw image but also needs to be protected such that the original image cannot be reconstructed even if the biometric data is available. The transforming of raw biometric data such as face image should not decrease the comparison performance limiting the use of biometric services. It can therefore be deduced that the feature representation and the template protection scheme should be robust to have reliable smartphone biometrics. This chapter presents two variants of a new approach of template protection by enforcing the structure preserving feature representation via manifolds, followed by the hashing on the manifold feature representation. The first variant is based on the Stochastic Neighbourhood Embedding and the second variant is based on the Laplacian Eigenmap. The cancelability feature for template protection using the proposed approach is induced through inherent hashing approach relying on manifold structure. We demonstrate the applicability of the proposed approach for smartphone biometrics using a moderately sized face biometric data set with 94 subjects captured in 15 different and independent sessions in a closed-set scenario. The presented approach indicates the applicability with a low Equal Error Rate,

This chapter is an extended version of our earlier work [1].

K. B. Raja (✉)
University of South-Eastern Norway, Kongsberg, Norway
e-mail: kiran.raja@usn.no; kiran.raja@ntnu.no

K. B. Raja · R. Raghavendra · M. Stokkenes · C. Busch
Norwegian Biometrics Laboratory, NTNU, Gjøvik, Norway
e-mail: raghavendra.ramachandra@ntnu.no

M. Stokkenes
e-mail: martin.stokkenes@ntnu.no

C. Busch
e-mail: christoph.busch@ntnu.no

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_15

$EER = 0.65\%$ and a Genuine Match Rate, $GMR = 92.10\%$ at False Match Rate (FMR) of 0.01% for the first variant and the second variant provides $EER = 0.82\%$ and $GMR = 89.45\%$ at FMR of 0.01%. We compare the presented approach against the unprotected template performance and the popularly used Bloom filter template.

15.1 Introduction

The use of biometrics as an authentication mechanism for a number of secure access services such as banking, border control or civilian identity management has resulted in the popularity of new generation biometric sensors in devices such as smartphones. A number of real-world applications using a smartphone for biometric authentication have demonstrated the success for corporations and convenience to the customers [2, 3]. Complementing the success from industry, there have been a number of academic works investigating various aspects of biometrics usage on the smartphone. A set of works have investigated the use of face biometrics [4, 5], periocular biometrics [6] and a few on the iris biometrics [4, 7]. Another set of works have indicated the use of a multi-modal approach for smartphone authentication to compensate the performance losses due to non-standard biometric data on smartphone [4, 5, 7]. While the use of biometrics provides versatility and convenience, the challenge of storing the biometric data on smartphones is not addressed to a greater extent. Unlike the password mechanisms, the original biometric characteristics are limited (one face, two irises, ten fingerprints) and thus cannot be replaced for a user if compromised, especially if the smartphone with biometric data is stolen or lost making the data available to maligned parties. It is therefore essential to store the biometric data in a protected manner such that the original biometric image (e.g. face image) cannot be reconstructed under the loss of a smartphone, leading to a need for *irreversibility*.

As an impact of protecting the biometric data, one can expect performance degradation in biometric authentication as the protected templates are typically a result of a number of transformations which may suffer a loss of information [8, 9]. The loss in biometric performance implies either rejecting the genuine subject repeatedly (corresponding to false reject rate—FAR) or accepting the subjects falsely (corresponds to false accept rate—FAR). While it is desired to have FAR and FRR simultaneously at very low values in an ideal biometric system, it is at least expected to prevent no false accepts in practical application with minimal possible false rejects, especially in the use case such as personalized banking applications to prevent monetary loss [2, 10]. The template protection schemes for smartphones thus need to consider performance factor and maintain the performance as equivalent to performance without template protection, or better performance than no-template protection. Given that smartphone is a personal device, it can be generalized that the same device is used to access a number of different services by the user. A direct implication of using the same biometric data (e.g. face) also enforces the need to make the biometric template *unlinkable* between different services from both user and the service provider perspective.

Although the requirements of template protection have been laid out in ISO-24745 [11], there are not many works reported on the smartphone biometric template protection. In this chapter, we present a new approach of protecting biometric templates on the smartphone by exploiting the feature space and using it to the advantage of creating protected templates. Specifically, we employ structure preserving manifold representation to keep the relational features intact prior to creation of the template. The creation of a protected template itself is based on the hashing approach to derive robust representation. While hash-based representation aids in deriving secure transformed template, there are a number of practical considerations in obtaining a stable hash for biometric data.

- The biometric data (e.g. face or fingerprint) varies across different captures, different sessions, different capture conditions and different camera/smartphones. The change in the captured biometric data under these conditions influences the biometric features proportionally. As a direct implication, the hash template representation will be impacted, resulting in lower biometric performance.
- The biometric features can provide high performance when the structural and relational neighbourhood features are preserved. For instance, minutia vicinity plays an important role in obtaining higher performance as compared to unordered fingerprint features. In a similar manner, one can argue that the features from the face can be highly reliable when the structural neighbourhood is preserved in the feature space.
- Our assertion is that hashing-based template protection can provide better performance if the extracted features preserve the neighbourhood and relational structure information making them stable against variations introduced due to capture process.

In this chapter, we present a new approach such that the structure of biometric features is preserved through the use of manifold representation and further use this representation to derive robust protected template via hashing. The proposed approach being computationally simple and efficient is suitable to be deployed on the low-power computational devices such as smartphones. Further, the cancelability is introduced by adopting an entropy-based sampling method to choose the features to obtain the manifold embedding. Through the properties of manifolds, i.e. inductive manifold and Stochastic Neighbourhood Embedding (SNE), we ensure the *irreversibility, unlinkability and revocability*. Further, the proposed approach is validated through the set of experiments on a moderate-sized database of 94 subjects with real biometric data captured using the smartphone. The key contributions of this chapter are:

1. A new approach for creating protected biometric templates is proposed which is based on the neighbourhood relation/structure preserving manifold representation of textural features and hash representation.
2. An experimental performance evaluation is presented to illustrate the validation of proposed approach through the use of smartphone biometric database. The proposed approach is compared against biometric performance of unprotected

template and Bloom filter-based protected template. All of our experiments correspond to the closed-set protocol as the work is addressed towards verification scenarios.
3. This chapter also presents a systematic discussion of the proposed approach for template protection and subsequently discusses unlinkability analysis. In the end, this chapter presents the merits and the limitations of the proposed approach to provide possible direction for future works.

In the rest of this chapter, Sect. 15.2 presents the related works followed by the Sect. 15.3 that discusses proposed approach for biometric template protection. The Sect. 15.4 provides the details on the employed database and the corresponding protocols for experiments. The experiments and the obtained results are discussed in Sect. 15.5 along with the brief discussion on unlinkability analysis in the Sect. 15.5.2 to demonstrate the security level of the proposed template protection method. A set of concluding remarks and a list of potential future work is provided in Sect. 15.6.

15.2 Related Works

A number of approaches can be adopted to deal with this problem of biometric template protection[11] which are either cancellable biometrics or biometric cryptosystems [8, 9, 12–17]. In this work, we adopt the *template protection approach through cancellable biometrics*. The goal of cancellable biometrics is to derive a biometric template that is irreversibly distorted while keeping the uniqueness for all biometric purposes such as identification and verification. Cancellable biometrics can be achieved through methods from simple mathematical transformations to approaches based on hashing. In this work, we adopt hashing-based template protection scheme with a set of key constraints to fulfil the properties required for biometric template protection while still achieving high biometric accuracy in a protected domain biometric comparison [1].

15.3 Proposed Approach for Protected Biometric Templates

The proposed approach of protected template creation is presented in Fig. 15.1. As depicted in Fig. 15.1, the features from biometric data are first extracted using texture descriptors. Specifically, we utilize widely employed Binarized Statistical Image Features (BSIF). The set of extracted features are represented using the manifold representation such that the neighbourhood representation is preserved. Given the set of enrolment images for the subjects, the proposed approach derives the hash projection matrix from the manifold representation of features. Through the projection matrix, we create the protected templates for each subject in the enrolment set. In a similar

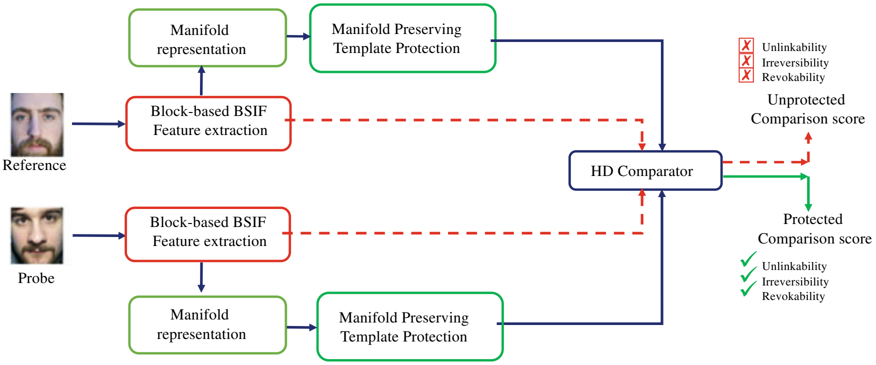

Manifold representation
Manifold Preserving Template Protection
☒ Unlinkability
☒ Irreversibility
☒ Revokability
Unprotected Comparison score
Block-based BSIF Feature extraction
Reference
HD Comparator
Block-based BSIF Feature extraction
Probe
Protected Comparison score
✓ Unlinkability
✓ Irreversibility
✓ Revokability
Manifold representation
Manifold Preserving Template Protection

Fig. 15.1 Biometric system with proposed template protection is indicated by the path followed by solid lines which also satisfies the properties of biometric template protection

manner, when a verification attempt is made by the subject, the textural features are extracted using BSIF followed by the manifold representation. The features are then projected using learnt hash projection matrix to derive the protected template representation for probe data. The templates in the protected domain for both enrolment and the probe are compared using a simple Hamming distance measure to establish the biometric performance. The details of each component of the proposed approach are presented in the section below.

15.3.1 Feature Vector from Binarized Statistical Image Features

Given the preprocessed biometric image, we first extract the textural descriptors using the Binarized Statistical Image Features (BSIF)[18]. The descriptors are obtained by convolving the image with the set of filters in the BSIF filter bank which is learnt using the independent component analysis of natural image patches. The choice of BSIF filters to extract the descriptors is motivated by high biometric performance reported in many earlier works [5, 18, 19]. Further, to make the descriptors highly unique, we employ both block-based feature extraction and multi-scale representation through the use of a number of filters from BSIF. Specifically, we employ the filters that correspond to 3×3, 5×5, 7×7, 9×9, 11×11, 13×13, 15×15 and 17×17 pixels with eight orientations. The pixel-wise response from the convolution of different orientation filters within a chosen filter is combined to obtain a final response through the thresholding and binarization approach such that a value between $0 - -255$ is obtained for every pixel [18]. The extracted features are further represented using histogram representation in the subsequent steps. Further, the uniqueness of the features from biometric images is enhanced through block-based approach where prior to extracting the BSIF features, each image is divided into a

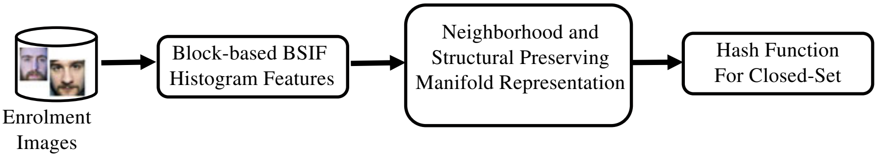

Enrolment Images
Block-based BSIF Histogram Features
Neighborhood and Structural Preserving Manifold Representation
Hash Function For Closed-Set

Fig. 15.2 Schematic representation of learning hash function for closed enrolment set

number of blocks and BSIF features extracted thereon. In this chapter, we employ 32 blocks of size 8 × 20 pixels from a resized biometric image of size 64 × 80 pixels. The set of all resulting histograms are concatenated to form a feature vector. The feature vector is further binarized using a simple zero thresholding [20, 21] to form a binary feature vector such that Hamming distance can be easily employed to derive biometric performance. Figure 15.2 presents the number of steps involved in extracting the final feature vector in this chapter.

15.3.2 Structure Preserving Biometric Feature Representation and Template Protection

As argued in the introduction, preserving the neighbourhood structure results in better biometric performance and thus in this section, we discuss the approach for preserving structure and neighbourhood within the feature vector of biometric data. Learning compact and effective hash codes can be achieved through embedding the original data into a low-dimensional space while simultaneously preserving the inherent neighbourhood structure [22]. In the line of the same argument, a set of works have demonstrated that nonlinear manifold learning methods are more powerful than linear dimensionality reduction techniques as they can effectively preserve the local structure of the input data without the explicit knowledge of global linearity [22, 23]. Motivated by such argument, we represent the features using the manifold representation using the t-Distributed Stochastic Neighbour Embedding (t-SNE) such that the structural relation of biometric data is preserved [23].

Given the binary feature vector Bx for a subject x within the set of enrolment samples, we attempt to learn the hash projection function and the details are provided herewith. The manifold representation for a given enrolment set $\mathbf{X}$ such that:

$$\mathbf{X} := \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$$

can be given by:

$$\mathbf{Y} := \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n\}$$

where $\mathbf{Y}$ is the manifold-based representation of corresponding binary feature vectors $\mathbf{X}$.

The key objective in deriving the manifold representation for the enrolment data q represented by $\mathbf{x}_q$ and i by $\mathbf{x}_i$ is that they should preserve both the neighbourhood and the structure in the original space. Thus, the problem can be formulated as a minimization problem that can be represented by Eq. 15.1.

$$min\{\sum_{i=1}^{n} \mathrm{w}(\mathbf{x}_q, \mathbf{x}_i)\|\mathbf{y}_q - \mathbf{y}_i\|^2\} \tag{15.1}$$

where $W_{qi} = w(\mathbf{x}_q, \mathbf{x}_i)$ is the affinity matrix as defined in [22]. Reformulating Eq. 15.1, it can be deduced that there exists $\mathbf{y}_q^\star$ where the solution of objective function is optimal such that:

$$\sum_{i=1}^{n} \mathrm{w}(\mathbf{x}_q, \mathbf{x}_i)(\mathbf{y}_q^\star - \mathbf{y}_i) = 0 \tag{15.2}$$

The assertion of the objective given in Eq. 15.1 is that the minimized distance between the points in embedding implies the distance between the nearest neighbours in the original dimension is preserved. For the sake of simplicity, we skip the details of each step, and the reader is referred to [1, 22].

Solving Eq. 15.2 and rearranging the terms, $\mathbf{y}_q^\star$ can be obtained as:

$$\mathbf{y}_q^\star = \frac{\sum_{i=1}^{n} \mathrm{w}(\mathbf{x}_q, \mathbf{x}_i)\mathbf{y}_i}{\sum_{i=1}^{n} \mathrm{w}(\mathbf{x}_q, \mathbf{x}_i)}. \tag{15.3}$$

Equation (15.3) is a simple formulation of manifold representation using the set of the linear combination of the features from the enrolment set [22].

Further, as the key properties of protected templates in biometrics need to fulfil *irreversibility, revocability and unlinkability* [11, 24, 25], we impose another condition to choose the sub-samples of the features via entropy-based selection to induce the first level of randomness. Given any manifold features $\mathbf{Y} \subseteq \mathbb{R}^r$ and $p \in \mathbb{N}$, the m-th entropy number $\varepsilon_m(\mathbf{Y})$ of Y is defined as

$$\varepsilon_m(Y) := \inf\{\varepsilon > 0 | \mathcal{N}(\varepsilon, Y, \| \cdot - \cdot \|) \leq m\} \tag{15.4}$$

where $\mathcal{N}$ is the covering number. Then, $\varepsilon_m(Y)$ is the smallest radius that Y can be covered by less or equal to m balls [22].

However, the challenge in realizing Equation (15.4) is the difficulty to cover all the wide range $\mathbf{Y}$ and therefore, an alternative possibility would be to use m clusters to cover $\mathbf{Y}$ where the clustering can be performed by *K-means* algorithm. The cluster centres are required to have the largest overall weight with respect to the points from their own cluster, i.e.

$$\sum_{i \in I_j} \mathrm{w}(\mathbf{c}_j, \mathbf{x}_i)$$

indicating the cluster centres as expressed by $\hat{\mathbf{y}}_q$. Using the relation mentioned above, Eq. (15.3) can be written as given by Eq. 15.5 along with the sign function, which translates to hash function. The hash function obtained by binarizing the low-dimensional embedding not only preserves the manifold with neighbourhood but also provides the binary templates [22].

$$h(\mathbf{x}) = \mathrm{sgn}\left(\frac{\sum_{j=1}^{m} \mathrm{w}(\mathbf{x}, \mathbf{c}_j)\mathbf{y}_j}{\sum_{j=1}^{m} \mathrm{w}(\mathbf{x}, \mathbf{c}_j)}\right) \tag{15.5}$$

where $\mathrm{sgn}(\cdot)$ is the sign function and

$$\mathbf{Y_B} := \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_m\}$$

is the embedding for the base set

$$\mathbf{B} := \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_m\}$$

which is the cluster centres obtained by K-means.

The approach formulated by Eq. 15.5 is the manifold representation (a.k.a., embedding) for the enrolment data:

$$\mathbf{Y} = \bar{\mathbf{W}}_{\mathbf{XB}}\mathbf{Y_B}, \tag{15.6}$$

where $\bar{\mathbf{W}}_{\mathbf{XB}}$ is defined using the cluster centres:

$$\bar{\mathbf{W}}_{ij} = \frac{\mathrm{w}(\mathbf{x}_i, \mathbf{c}_j)}{\sum_{i=1}^{m} \mathrm{w}(\mathbf{x}_i, \mathbf{c}_j)} \tag{15.7}$$

for $\mathbf{x}_i \in \mathbf{X}$, $\mathbf{c}_j \in \mathbf{B}$.

In this chapter, we employ two different approaches to derive a manifold representation of the enrolment features. The first approach to derive manifold representation is through Stochastic Neighbourhood preserving Embedding (t-SNE) [23] as proposed in our recent work. It was shown in the preliminary work in [1] that t-SNE based structure preserving manifold is able to preserve both biometric features as well as performance. While in the first approach, the manifold representation and the hashed projection is only based on the optimization of one function given in Eq. 15.5, we explore another similar approach proposed in [22] with a set of relaxations to consider features in the manifold representation and the features in the original space.

$$min\{\sum_{i=1}^{n} \mathrm{w}(\mathbf{x}_i, \mathbf{x}_j)\|\mathbf{y}_i - \mathbf{y}_j\|^2 + \lambda \sum_{\mathbf{x}_i \in \mathbf{Y}, \mathbf{x}_j \in \mathbf{x}}^{n} \mathrm{w}(\mathbf{x}_i, \mathbf{x}_j)\|\mathbf{y}_i - \mathbf{y}_j\|^2\} \tag{15.8}$$

where λ is the relaxation parameter. Through the specific reformulation provided in [22], Eq. 15.8 can be presented as the Laplacian Eigenmap. The first approach relies on t-SNE and is referred to as ***Manifold-structure Preserving Biometric Template (MaPBiT)*** in our earlier work. The second approach relies on Laplacian Eigenmap (LE), and we hereby refer to it as ***Manifold-structure Preserving via Laplacian Eigenmap for Biometric Template (MaPLEBiT)***.

15.4 Data set and Evaluation Protocol

This section provides the details of the data set employed for the experimental evaluation of the approach presented in this chapter. We employ a face data set captured from smartphone that consists of images corresponding to 94 unique subjects [5, 21]. The composition of the images in the database is provided in Table 15.1.

The images are captured in 15 different attempts where 5 captures correspond to the high-quality enrolment samples and 10 correspond to the probe attempts under varying capture conditions such as illumination and background. We retain the original partition of the database where the complete data set is partitioned to *Development* and *Testing/Evaluation*. The *Development* set consists of data captured from 21 different subjects, while the *Testing* set consists of data captured from 73 subjects. The parameters of experiments such as number of filters, size of filters and hashing features are selected on the basis of empirical trials on the *Development* data set. The selected parameters are used for experiments on the *Testing* data set to report the results in this work.

15.4.1 Evaluation Protocols

This section outlines the experimental protocols followed in this chapter. We adopt the protocols corresponding to the earlier works [21] that have 5 images in enrolment set and 10 images in the probe set. The results are presented in the terms of Equal Error Rate (EER %) such that a symmetrical error distribution of False Match Rate (FMR) versus False Non-Match Rate (FNMR) can be visualized. The error rates are accompanied by the Detection Error Trade-off (DET) curves to understand the algorithmic performance.

Table 15.1 Statistics of the smartphone face biometric data set

	Development data set	Testing data set
Subjects	21	73
Device	Samsung Galaxy S5	Samsung Galaxy S5
Reference images	5	5
Probe images	10	10
Genuine comparisons	1050	3650
Impostor comparisons	21,000	262,800

15.5 Experiments and Results

Along with the results from the proposed approach in this chapter, we present the results from two other approaches that correspond to the performance from unprotected biometric templates and another set corresponding to protected templates through Bloom filter approach. A significant difference to be noted in the experimental protocols is that while the unprotected and protected template performance is independent of enrolment samples, the proposed approach relies on the known enrolment set (closed-set biometric scenario).

Unprotected Templates: In order to provide biometric performance, we provide the baseline evaluation with unprotected biometric templates using the multi-scale block-based Binarized Statistical Image Features (BSIF) which are derived using a set of varying filters of size such as 3×3, 5×5, 7×7, 9×9, 11×11, 13×13, 15×15 and 17×17 with each of basis size corresponding to eight layers. Further, the biometric face image is partitioned into 32 different blocks of size 8×20 pixels as discussed earlier in Sect. 15.3.1. The resulting concatenated histograms are binarized using simple zero thresholding, and the distance between two histograms is measured using Hamming distance in the unprotected domain.

Bloom filter Template Protection: In a similar manner to unprotected templates, we employ Bloom filter representation to derive protected templates using the features as discussed in Sect. 15.3.1. Further, Hamming distance is employed to measure the dissimilarity between the protected templates to derive the biometric templates [21].

Proposed Template Protection Schemes: In order to evaluate the proposed approaches, we adopt features as outlined in Sect. 15.3.1. As the features are further represented in binary format, we employ simple Hamming distance to derive the biometric performance. The key difference here compared to unprotected and Bloom filter based template protection is the number of filters employed. *In the proposed approach, we employ block-based approach with only* 9×9 *pixels with* 8 *bits while both unprotected and Bloom filter based templates employ* 8 *different filters along with block-based approach.*

15.5.1 Discussion on Results

As it can be observed from Table 15.2 and Fig. 15.3, the proposed approach (both variants) provide better performance with respect to both FMR and FNMR. The better results compared to unprotected templates can be fully attributed to the optimization procedure in selecting the unique bits for the hash. *While one can argue that the performance is primarily due to optimization from the known set, it can be counter-argued that data from pseudo-users can be used to derive the templates for each user of the smartphone.* Given this argument, we are justified in using this approach to obtain the performance close to unprotected templates. The obtained results have

Table 15.2 Results obtained for unprotected templates, Bloom filter template & proposed template protection (MaPBiT and MaPLEBiT). Genuine Match Rate (GMR) reported at False Match Rate of 0.01%. The results with ± presents the average variance over a number of experimental evaluation

Template	Face	
	EER	GMR
Unprotected-MBSIF	1.65	90.05
Protected-Bloom filter	2.91	82.68
Protected-Proposed-MaPBiT	0.65 ± 0.18	92.10 ± 0.78
Protected-Proposed-MaPLEBiT	0.82 ± 0.12	89.45 ± 0.57

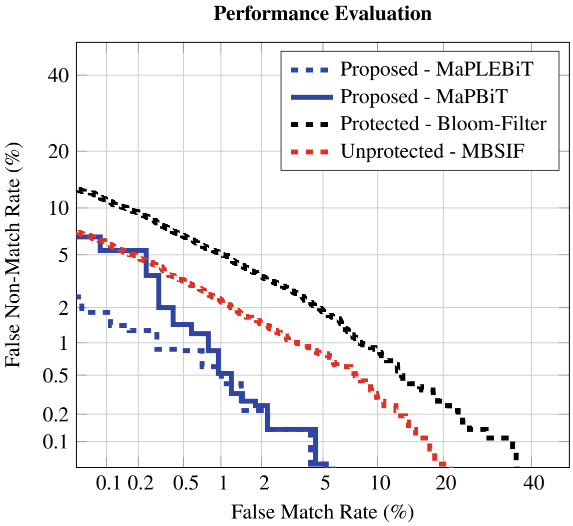
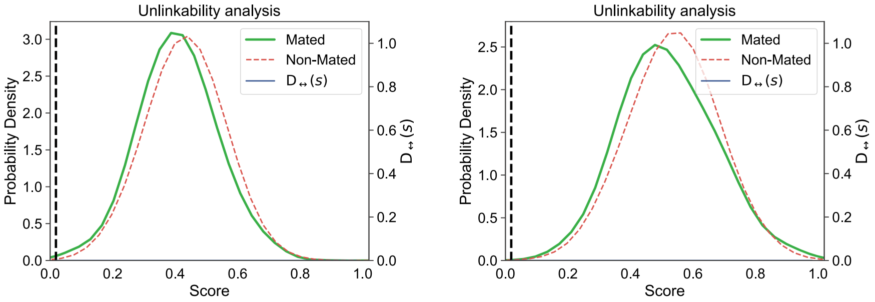


Fig. 15.3 Comparison of biometric performance using DET for smartphone face biometric data set

validated our intuition that retaining the inherent structural similarity of biometric features via neighbourhood preserving embedding improves the protected template performance. Further, the results also suggest that the approach can be used in two different variants with t-SNE and Laplacian Eigenmap based manifolds.

15.5.2 Unlinkability Analysis

This section presents the unlinkability analysis of the proposed approach through the metric proposed in [12, 26]. Here, it is assumed that the same biometric system is deployed for two different applications, and it should not be possible to tell if an individual present in one is also present in the other. The biometric templates from the same individual (one template from each application) are compared to generated



(a) MaPLEBiT (Laplacian Eigenmap)

(b) MaPBiT (tSNE)

Fig. 15.4 Unlinkability analysis of proposed template protection

mated score distribution. Similarly, the biometric templates from different individuals are compared to generate the non-mated score distribution. Greater overlap between the two distributions demonstrates greater unlinkability.

The score distributions of two cancellable templates are presented in Fig. 15.4 for two variants of the proposed approach. As observed from Fig. 15.4, both the variants, *MaPBiT* and *MaPLEBiT*, demonstrate a good degree of unlinkability. This can be interpreted through the genuine and the imposter distribution which have a high degree of overlap indicating the low probability of linkability.

15.5.3 Limitations of Current Work and Potential Future Works

The proposed approach in both variants has demonstrated not only good biometric performance but also the applicability for the smartphone biometric scenario. While the performance closely matches the unprotected template biometric performance, the proposed approach inherently needs known enrolment set. Although this limitation can be addressed through employing a set of pseudo-users, real-time analysis with large-scale biometric data needs to be conducted. As a second advantage, the proposed approach results in a compact template size which can be aptly used in smartphone biometric scenario demanding very low memory size.

15.6 Conclusions

In this chapter, an approach for biometric template protection for smartphone data was presented with two variants. The need for preserving the sensitivity of the biometric data while respecting key properties of *irreversibility*, *unlinkability* and *renewability* has been met through the proposed approach. The chapter has systematically argued the use of manifold preserving feature representation to improve the biometric performance of template protection. The argument has been well illustrated using the two variants of manifold representation with an experimental analysis of the proposed approach. The results obtained on a moderate-sized face biometric database indicate the applicability of proposed approach with a resulting accuracy of $EER \approx 0.65\%$ for the first variant (t-SNE) and the $EER \approx 0.82\%$ for the second variant (Laplacian Eigenmap), both of which are better than the EER (1.65%) of the unprotected biometric system. Unlinkability analysis of the proposed approach has shown very low chance of linkage issues and thereby providing the better cancellable biometric templates in a closed-set scenario.

Acknowledgements This work was partially carried out under the funding of the Research Council of Norway (Grant No. IKTPLUSS 248030/O70).

References

1. Raja KB, Raghavendra R, Busch C (2018) Manifold-structure preserving biometric templates - a preliminary study on fully cancelable smartphone biometric templates. In: Proceedings of the ICME, pp 1–8
2. ZOLOZ Real ID. http://www.zoloz.com, 2017. Accessed on 01 Jan 2018
3. Salesky J (2017) Providing a frictionless banking experience: What banks can learn from apple. Am Bankers Assoc. ABA Bank J, 109(1):38–38,50. Copyright - Copyright Naylor, LLC Jan/Feb 2017; Last updated - 2017-02-07; CODEN - ABAJD5
4. De Marsico M, Galdi C, Nappi M, Riccio D (2014) Firme: face and iris recognition for mobile engagement. Image Vis Comput 32(12):1161–1172
5. Raja KB, Raghavendra R, Stokkenes M, Busch C (2015) Multi-modal authentication system for smartphones using face, iris and periocular. Proceedings of 2015 international conference on biometrics, ICB 2015, pp 143–150
6. Rattani A, Derakhshani R, Saripalle SK, Gottemukkula V (2016) Icip 2016 competition on mobile ocular biometric recognition. In: 2016 IEEE international conference on image processing (ICIP), pp 320–324
7. Raja KB, Raghavendra R, Vemuri VK, Busch C (2015) Smartphone based visible iris recognition using deep sparse filtering. Pattern Recognit Lett 57:33–42
8. Ratha NK, Chikkerur S, Connell JH, Bolle RM (2007) Generating cancelable fingerprint templates. IEEE Trans Pattern Anal Mach Intell 29(4):561–572
9. Ratha NK, Connell JH, Bolle RM (2001) Enhancing security and privacy in biometrics-based authentication systems. IBM Syst J 40(3):614–634
10. Council of European Union. Apple touch id, Accessed on April 2016. http://www.cbsnews.com/news/should-you-fear-apples-fingerprint-scanner/
11. ISO/IEC JTC1 SC27 Security Techniques. ISO/IEC 24745:2011. information technology - security techniques - biometric information protection, 2011
12. Gomez-Barrero M, Rathgeb C, Li G, Ramachandra R, Galbally J, Busch C (2018) Multibiometric template protection based on bloom filters. Inf Fusion 42:37–50
13. Jutta H-U, Elias P, Andreas U (2009) Cancelable iris biometrics using block re-mapping and image warping. In ISC, vol 9. Springer, pp 135–142
14. Jin ATB, Ling DNC, Goh A (2004) Biohashing: two factor authentication featuring fingerprint data and tokenised random number. Pattern Recognit 37(11):2245–2255

15. Patel VM, Ratha NK, Chellappa R (2015) Cancelable biometrics: a review. IEEE Signal Process Mag 32(5):
16. Pillai JK, Patel VM, Chellappa R, Ratha NK (2011) Secure and robust iris recognition using random projections and sparse representations. IEEE Trans Pattern Anal Mach Intell 33(9):1877–1893
17. Uludag U, Pankanti S, Prabhakar S, Jain AK (2004) Biometric cryptosystems: issues and challenges. Proc IEEE 92(6):948–960
18. Juho K, Esa R (2012) BSIF: binarized statistical image features. 21st ICPR (Icpr):1363–1366
19. Raja KB, Raghavendra R, Busch C (2014) Binarized statistical features for improved iris and periocular recognition in visible spectrum. In: 2014 International Workshop on Biometrics and forensics (IWBF), pp 1–6
20. Stokkenes M, Ramachandra R, Raja KB, Sigaard M, Gomez-Barrero M, Busch C (2016) Multi-biometric template protection on smartphones: an approach based on binarized statistical features and bloom filters. In: Iberoamerican Congress on Pattern Recognition. Springer, pp 385–392
21. Stokkenes M, Ramachandra R, Sigaard MK, Raja KB, Gomez-Barrero M, Busch C (2016) Multi-biometric template protectiona security analysis of binarized statistical features for bloom filters on smartphones. In 6th IPTA. IEEE 2016:1–6
22. Shen F, Shen C, Shi Q, Van Den Hengel A, Tang Z (2013) Inductive hashing on manifolds. In: 2013 IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 1562–1569
23. van der Maaten LJP, Hinton GE (2008) Visualizing high-dimensional data using t-sne
24. Marta G-B, Christian R, Javier G, Christoph B, Julian F (2016) Unlinkable and irreversible biometric template protection based on bloom filters. Inf Sci
25. Hermans J, Mennink B, Peeters R (2014) When a bloom filter is a doom filter: security assessment of a novel iris biometric template protection system. In: 2014 International conference of the biometrics special interest group (BIOSIG), pp 1–6
26. Gomez-Barrero M, Galbally J, Rathgeb C, Busch C (2018) General framework to evaluate unlinkability in biometric template protection systems. IEEE Trans Inf Forensics Secur 13(6):1406–1420

Chapter 16
Security, Privacy, and Usability Challenges in Selfie Biometrics

Mikhail Gofman, Sinjini Mitra, Yu Bai and Yoonsuk Choi

Abstract From biometric image acquisition to matching to decision making, designing a selfie biometric system is riddled with security, privacy, and usability challenges. In this chapter, we provide a discussion of some of these challenges, examine some real-world examples, and discuss both existing solutions and potential new solutions. The majority of these issues will be discussed in the context of mobile devices, as they comprise a major platform for selfie biometrics; face, voice, and fingerprint biometric modalities are the most popular modalities used with mobile devices.

16.1 Introduction

Modern mobile devices support face, voice, fingerprint, and iris recognition. These biometric systems operate under uncontrolled conditions; they must contend with security threats of fake biometrics; they must protect against the divulgence of biometric templates if the device is lost or stolen; and they are constantly pressured to be user-friendly. In this chapter, we will provide an overview of security and usability challenges and solutions in mobile biometric systems. Special focus will be placed on issues of security attacks involving fake biometrics, template security, and the

M. Gofman (✉)
Department of Computer Science, California State University, Fullerton,
800 N State College Blvd, Fullerton, CA 92831, USA
e-mail: mgofman@fullerton.edu

S. Mitra
Department of Information Systems and Decision Sciences, California State
University, Fullerton, 800 N State College Blvd, Fullerton, CA 92831, USA
e-mail: smitra@fullerton.edu

Y. Bai · Y. Choi
Department of Computer Engineering, 800 N State College Blvd, Fullerton, CA 92831, USA
e-mail: ybai@fullerton.edu

Y. Choi
e-mail: yochoi@fullerton.edu

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision
and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_16

making of mobile biometric systems user-friendly. The chapter concludes with the discussion of case studies concerning selfie biometric systems.

16.2 Security Issues Overview

The iPhone 5s was among the first commercially successful consumer mobile devices that supported fingerprint recognition [1]. Within a week of its release (2013), Chaos Computer Club (CCC), a German hacker group, had bypassed the fingerprint sensor "using easy everyday means." According to the CCC website:

> The fingerprint of the enrolled user is photographed with 2400 dpi resolution. The resulting image is then cleaned up, inverted and laser printed with 1200 dpi onto transparent sheet with a thick toner setting. Finally, pink latex milk or white wood glue is smeared into the pattern created by the toner onto the transparent sheet. After it cures, the thin latex sheet is lifted from the sheet, breathed on to make it a tiny bit moist and then placed onto the sensor to unlock the phone. [2]

Since then, methods were identified that could both bypass fingerprint recognition on later models of Apple iPhone [3]—and on Android-based devices [4]—and defeat face recognition systems on more modern devices, such as the iPhone X (released in 2017) [5, 6]. Attacks of this type are becoming progressively more sophisticated and effective as hackers continue to develop new methodologies. As an increasing number of users continue to ditch passwords and pin codes in favor of selfie biometric systems as their primary security gatekeepers, it is critical that these systems remain resilient to security attacks.

In addition to the security threats posed by fake biometric attacks, there exist concerns about the security and privacy of the data in the biometric templates. A template is a digital representation of the user's identifying features that are created from biometric samples initially supplied by the user when he/she sets up his/her device. The samples provided at the time of authentication are then matched against the stored template. If a template is divulged—say, if the device is lost, stolen, or hacked—hackers can use the template data to bypass biometric systems that use the same biometric modality.

The consequences of stolen biometric data are exacerbated by the fact that biometric modalities cannot be as easily changed as passwords can. Moreover, the stolen template data can be used for surveillance purposes in order to track users while they use the compromised biometric in different places and at different times.

These security concerns prompted mobile device manufacturers and security researchers to develop various software- and hardware-based defenses. To give the reader a better grasp of these security challenges and solutions, we begin with a generic threat model applicable to all biometric systems. We then focus on developing solutions to defend against trait-spoofing attacks and protect templates in mobile devices.

16.3 The Threat Model

The model depicted in Fig. 16.1 is adapted from Ratha et al. [7].

In this model, the *sensor* is used to acquire the raw biometric data. Next, the *feature extractor* module extracts the identifying information from the raw data. The features are then matched against the templates of enrolled users by the *matcher* module. Finally, the matcher outputs a "yes/no" decision as to whether the sample supplied during authentication matches the stored template.

The system in Fig. 16.1. The selfie biometric system threat model is susceptible to the following threats:

1. The attacker can place or present a *fake biometric* on the sensor or in front of the camera (e.g., a fake finger or a photograph of a face) in order to result in a false positive identification of an illegitimate subject. Multiple such attacks have proven to be successful against mobile devices [3–5].
2. Raw biometric data from the sensor can be *recorded* and *replayed* such that the attacker may gain access to the system (e.g., voice samples).
3. A *feature extractor may be replaced* by the attacker with another extractor that generates a predetermined set of features.
4. *Features extracted from biometric data can be replaced* with some other features chosen by the attacker.
5. A *trait-matching algorithm can be replaced* with the attacker's own matching algorithm.
6. *Biometric templates can be accessed by and tampered with* by the attacker. This includes insertion, deletion, modification, or theft of the templates.
7. *The retrieval of the template from the template database can be compromised*; for example, the attacker can replace the template retrieved from the requested user with his/her own template.

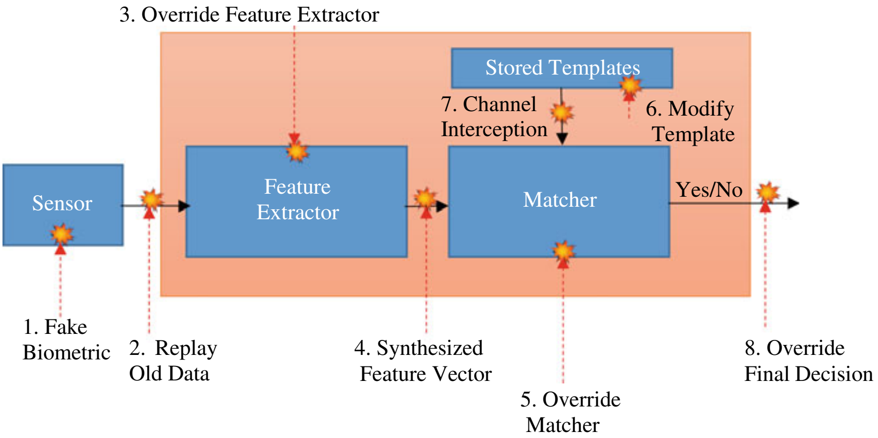


Fig. 16.1 The selfie biometric system threat model

8. *The decision of the identity verification system can be overridden* by the attacker such that he/she may gain unauthorized access to the system or deny access to a legitimate user.

Mobile device manufacturers have developed specialized computing hardware that helps mitigate attacks (2)–(8), such as the Apple Corporation's Secure Enclave processor [8] used with iPhones. This hardware is physically isolated from the main computing architecture of the device. Therefore, even if the applications, operating system, and the primary computing hardware are compromised, the function of the biometric system remains unaffected.

To further mitigate an attack (8), the templates store an output of a one-way function computed from the original biometric features. This output can still be used to match the features while rendering the derivation of the original identifying features computationally difficult or impossible. This is an important consideration in mobile devices because, even if the data is stored on a physically isolated, tamperproof hardware chip, attackers can disassemble a lost or stolen device in an attempt to bypass tamperproof hardware security mechanisms and thus retrieve the data.

Attack (1) remains an important concern. Some recent, noteworthy compromises of selfie biometric systems include:

- According to The Verge, "All it took was some dental mold to take a cast, some play-dough to fill it, and then a little trial and error to line up the play-dough on the fingerprint reader. We did it twice with the same print: once on an iPhone 6 and once on a Galaxy S6 Edge" [9].
- According to MyBroadband: "[When] our new gelatin [cast of person's fingerprints], was placed on the Nokia 5's sensor, the result was almost instant—the device was unlocked" [10].
- iPhone X's Face ID face recognition was bypassed by a Vietnamese company who created a 3-D mask of the individual's face that the device recognized as the face of the legitimate user [5].
- There were reports of children using their faces to unlock their parents' locked iPhone Xs using Face ID because children's faces may be sufficiently similar to their parents' faces [11].
- There were reports and demonstrations of iPhone X being unlocked by people who did not look alike [12].
- Banking mobile applications based on face recognition have been bypassed using a pre-recorded video of the user's face [13].

To help combat these and similar attacks, researchers and mobile device manufacturers have developed more robust sensors and additional hardware-based liveness testing techniques. These techniques help ensure the biometric reading from the sensor is indeed given by a living human being (e.g., checking the finger's pulse during fingerprint recognition and requiring eye blinking during face recognition).

A variety of software-based data processing techniques for detecting spoofed biometrics have also been proposed. Some have focused on frustrating attacks directed at specific modalities (e.g., face, voice, and fingerprint), while some proposed recognizing people based on multiple biometric modalities in order to challenge the attacker to falsify more than one modality. Although these measures are useful, experience implies that all measures are likely to be eventually defeated by the attackers—the question is not “if,” but rather, “when.” Regardless, continuous innovation in technologies and methods for detecting fake biometrics is a practical necessity.

Next, we discuss attacks (1) and (7) along with their countermeasures. In selfie biometrics, these particular attacks have proven to be the most widely executed in practice, the most widely discussed in the literature, and the utmost focus of public concern.

16.3.1 Presentation Attacks

Throughout this section, we use the ISO/IEC 30107 standard definition of a “presentation attack”—“presentation of an artifact or human characteristic to the biometric capture subsystem in a fashion that could interfere with the intended policy of the biometric system” [14]—to refer to attacks involving falsified biometrics.

Presentation attacks have been a concern in biometric systems since the field’s inception. An attacker can utilize knowledge of, for example, a user’s fingerprints in order to fabricate a fake finger that he/she can then apply to the sensor and foil the system. Similarly, an attacker can hold up a photograph of the user’s face before the camera in an attempt to unlock a mobile device that uses face recognition as a gatekeeper. Biometric researchers, manufacturers, and standardizing groups (e.g., International Organization for Standardization Standards Office and National Institute of Standards and Technology [NIST]) are currently working to develop efficient methodologies to stop such attacks. With the increasing reliance on mobile biometrics in government applications, NIST developed a protocol to ensure security in mobile device biometric applications [15]. Selfie biometric systems in mobile devices have added a sense of urgency to these efforts; although proven vulnerable, millions of consumers and organizations continue to rely on the security afforded by face and fingerprint recognition on their mobile devices.

Protecting devices against presentation attacks is challenging. First, additional hardware may be needed to allow biometric sensors to differentiate between a real biometric and a spoof. The increased costs and design complexity are problematic for mobile selfie biometric systems wherein strict size and cost constraints pose issues. Second, many mobile devices have limited computational resources, which precludes the use of the best available software approaches that can be computationally intensive.

Next, we discuss presentation attacks and countermeasures for face, fingerprint, and voice modalities that are commonly used in mobile device biometrics.

16.3.2 Face Presentation Attacks

Face presentation attacks are classified based on whether they use 2-D or 3-D face artifacts. 2-D attacks typically involve the use of 2-D face videos, still photographs, and other forms of 2-D artifacts to deceive systems that identify people based on 2-D images (as many mobile selfie biometric systems do). 3-D attacks use 3-D masks and other types of 3-D artifacts to deceive face recognition systems based on 2-D or 3-D face images. According to a survey conducted by Galbally et al. [16] and Rattani and Derakhshani [17], the specific techniques in these categories can be summarized in the following ways.

(1) Photograph attacks: These involve presenting the camera with a 2-D face photograph of the legitimate user. The image can be printed on paper or displayed on the computer screen. This type of attack was successful against early versions of Android's face unlock feature [18] as well as modern Android-based devices (e.g., Samsung Galaxy S8 [18]).

If the system requires that the user blink, an animated image that mimics blinking can be created. Another technique involves creating a copy of the original face image, creating another image with the eyes erased, and then rapidly alternating these two images on the computer screen positioned in front of the camera. Such a technique was used to defeat the blinking detection mechanism in face unlock that was introduced in Android to counter presentation attacks [18].

Blinking can also be faked using a printed 2-D mask of the face with holes cut out for the eyes and mouth. The attacker then wears the mask in front of the camera and replicates natural blinking and mouth movements as needed [19].

(2) Video attacks: The attacker presents the camera with a video of the legitimate user's face. The video preserves a face's movements and texture and therefore can defeat rudimentary anti-spoofing mechanisms such as blinking [20]. This attack has been successful against mobile banking applications that utilize face recognition [19].

(3) 3-D mask attacks: Here, the attacker uses a 3-D mask of the legitimate user's face. Although the task of creating a 3-D mask is generally more difficult than finding a photograph or video of the legitimate user's face, the task is becoming easier due to the availability of 3-D printers capable of cheaply producing high-quality masks and services; for example, www.thatsmyface.com, for a current fee of $299 (at this point in time), can create a 3-D wearable mask from a 2-D face photograph. Another variation of a 3-D mask attack was used to bypass iPhone X's face recognition system based on 3-D imaging [21].

The research into 3-D mask attacks and defenses has recently accelerated due to the availability of datasets featuring different types of 3-D masks, such as 3-MAD [22].

Galbally et al. also discuss *feature-level dynamic*, *feature-level static*, *sensor-level*, and *score-level* approaches for defeating presentation attacks.

Feature-level dynamic approaches analyze the movements of the different face regions in order to detect a still 2-D photograph. The central idea is that the movements of the real face and the movements of the printed image will be different. They can also use challenge–response protocols requiring that users blink or make specific face gestures, such as smiling or turning. Although feature-level static techniques can help defeat attacks based on 2-D still photographs, they are less effective against video spoofs that contain natural movements. They do, however, make video spoofing attacks somewhat more difficult, as the attacker must find or fabricate a video of the victim performing a specific gesture.

More advanced feature-level countermeasures include comparing the movements of the foreground and background, implementing techniques that use local binary patterns (LBPs) [23] in order to track face movements or detect texture properties of a live face, analyzing face photographs taken in sequence in order to infer the 3-D structure of the face, and estimating the noise resulting from capturing the photograph.

Although Galbally et al. argue that face anti-spoofing, feature-level dynamic techniques require multiple face images during authentication and hence will not work in applications wherein a sequence of face photographs is unavailable, we believe this will not be a problem in the majority of mobile selfie biometric systems wherein such sequences can be readily captured from the device camera. However, if only one image is available, then the feature-level static approaches can be applied; these are generally faster yet tend to be less robust than their feature-level dynamic counterparts.

Feature-level static analysis techniques detect spoofed face images based on the single image rather than a sequence of images. Many techniques in this category are based on analyzing the texture of the face [24]. If a sequence of photographs or a video is available, then these techniques can be applied to individual photographs or video frames. The results of the analysis of each frame can then be fused together and the decision can be made based on the final score. This, however, is believed to be a less robust method than using the feature-level dynamic techniques described above.

Sensor-level techniques differ significantly from static and dynamic feature-level fusion techniques and typically require the integration of additional hardware into the sensor. These can include, for example, an extension of the sensing capabilities with the addition of the infrared or near-infrared (IR/NIR) cameras that capture information beyond the visible spectrum. Recent mobile devices such as iPhone X have recently started using special cameras that construct 3-D face models for face recognition [25], which help defeat spoofing using 2-D photographs and videos. The iPhone X Face ID camera projects IR rays onto 30,000 points of the face to construct a 3-D image of the face. A 2-D IR scan is also captured. According to an Apple white paper discussing Face ID [26]. "This data is used to create a sequence of 2-D images and depth maps" that are then used for authentication.

Although the Face ID system has already been bypassed, it thus far (at this point in time) appears to be more difficult to spoof [26] according to the many documented reports of failed spoofing attempts (i.e., 2-D photographs and videos, 3-D masks) that have worked against other devices. We, therefore, believe that combining IR and 3-D imaging techniques will certainly bring greater security to face recognition on mobile devices. The challenges of doing so require addressing the open research questions of optimally combining IR/NIR and 3-D data while coping with physical space, manufacturing costs, and computational constraints.

Score-level approaches employ different anti-spoofing strategies. Each strategy is implemented as a module that outputs a score indicating the likelihood that the face image is a spoof, and the scores from the different modules are then combined. The resulting score is then used to judge whether or not the image is a spoof.

Overall, we believe the most effective approaches will occur from combining static feature, dynamic feature, score-level, and sensor-level techniques; each technique possesses unique strengths. Advancements in mobile sensors and computing technologies are also expected to pave the way toward the development of new approaches to combat spoofing attacks. Furthermore, increased computing power capable of scaling increased computational loads imposed by the use of multiple presentation attack detection techniques will make the simultaneous implementation of multiple and simultaneous feature-level static, feature-level dynamic, sensor-level, and score-level techniques a possibility.

16.3.3 Fingerprint Presentation Attacks

Fingerprints are the most popular biometric [27]. Unlike, for example, the face or voice, fingerprints work well in poorly lit and noisy environments. At the same time, fingerprint recognition systems continue succumbing to presentation attacks. Some attacks are as simple as using various sticky materials to pick up a latent fingerprint from surfaces and then apply the captured print to the reader, while more sophisticated attacks include the use of 3-D printed fingers [28].

Marasco and Ross [29] published a survey documenting presentation attacks and proposed countermeasures. We use the survey to guide our discussion of the different types of attacks and countermeasures and then include remarks on their applicability in the mobile device context while discussing and analyzing modern works that specifically focus on mobile devices.

The attack types are categorized based on the methods used for faking a fingerprint:

Cooperative duplication: This occurs when the subject voluntarily presses his/her fingerprint into plaster or a similar material that captures the inverted impression of the fingerprint. The mold is then filled with some liquid material that later hardens and thus captures the actual impression of the fingerprint (e.g., gelatin).

This type of attack can be difficult to execute with mobile devices, as many users are unlikely to cooperate with the process. Indeed, when the authors of this chapter were constructing a multimodal biometric dataset constituting the face, ear, and fingerprints, nearly fifty volunteers were willing to donate their faces and ears. At the point in time this study was written, only a handful were willing to donate fingerprints.

Non-cooperative attacks: These types of attacks do not require the subject's cooperation and are a serious threat to mobile device fingerprint recognition. These can be divided into four sub-categories:

1. **Latent fingerprints**: When a finger touches certain surfaces (e.g., glass, metal, wood), it leaves a fingerprint impression. These impressions may then be collected and used for presentation attacks. Various techniques for collecting fingerprints have been developed [30] and are applicable for mobile fingerprint readers.
2. **Fingerprint re-activation**: When the finger contacts the sensor, it leaves a fingerprint. That fingerprint can be reactivated using techniques such as breathing on the sensor or applying graphite powder.

Earlier generations of fingerprint scanners, such as those used for the Samsung Galaxy S5 [31], required that the user swipe the finger across the sensor. Newer sensors, such as those used for the Samsung Galaxy S9, allow the user to press the finger onto the sensor and hold it in place. This is believed to be more user-friendly than swiping. However, since the swiping motion tends to wipe or at least distort the latent fingerprints (i.e., fingerprints left on the sensor surface from previous contact)—unlike pressing and holding—such an attack becomes a theoretically greater concern. More research is needed in order to establish the real extent of the threat.

3. **Cadaver**: This involves the use of a dead finger to unlock a device. According to multiple reports from law enforcement professionals, it is not uncommon for crime investigators to apply the fingers of corpses to the iPhone fingerprint reader in order to unlock the deceased person's device [32, 33]. These reports come in spite of the claims that anti-spoofing measures in the iPhone fingerprint sensors can successfully discriminate between a living and a dead finger [34].
4. **Fingerprint synthesis** is the use of the user's biometric template stored by the system in order to reconstruct the fingerprint. Such an attack inevitably requires access to the template. Apple and various Android-based mobile device manufacturers currently possess dedicated hardware and software that prevent the compromise of the template data that can frustrate this attack. The details of template security approaches will be discussed in the forthcoming sections.
5. **Other techniques**: Attackers can employ schemes to steal people's fingerprints—e.g., by leaving materials on surfaces often touched by people that capture fingerprints. Materials can include gel, plaster, or forensic fingerprint powders. The attacker can then later return to collect fingerprints.

A more sophisticated form of attack would involve secretly embedding a fingerprint scanner that produces high-resolution images of fingerprints or a device that captures the fingerprint topology in ATM machines and other places that frequently come into contact with human fingers. Indeed, a malicious mobile device manufacturer can choose to purposely leak fingerprint images from the user's phone back to the manufacturer, where they can then be used for presentation attacks.

A person's fingerprints can also be obtained through coercion or secretly without consent—e.g., pressing the finger into gel or plaster while the victim is asleep or distracted. The efforts and risks may well be worth the reward depending on the attacker's purpose.

Other potential, less orthodox threat vectors may exist as well. For example, it has been discovered that iPhone's Touch ID system allows the enrollment of pawprints of cats [35], dogs [36], and hedgehogs [37]. Some pet owners have allegedly used this technique to protect their phones (although we could not verify the validity of these accounts). Therefore, an attacker with access to the user's pet can replicate the pawprint using cooperative duplication techniques described above.

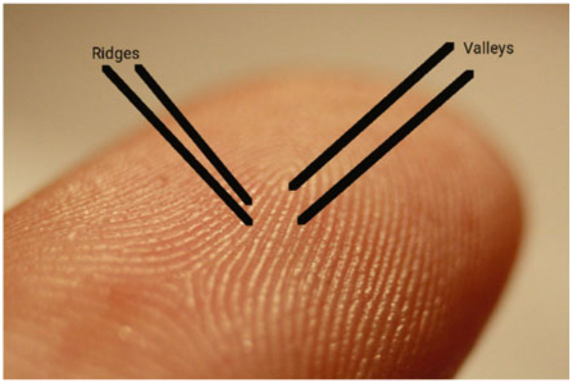
Next, we discuss techniques for countering the fingerprint presentation attacks. First, we provide an overview of the different types of defense measures and then discuss defense measures used with modern mobile devices.

Marsasco et al. [29] separated anti-spoofing techniques into two categories: hardware-based and software-based. Hardware-based measures require that additional anti-spoofing hardware be integrated into the sensor such that the sensor may discriminate between a live finger and a spoof. Software-based techniques process the biometric data and features in order to detect anomalies that can signal a spoofing attack. Such techniques can be broken down into dynamic and static techniques. Dynamic techniques include detecting ridge-based distortion and fingerprint perspiration properties. Static techniques include detecting anomalies in the finger texture, detecting the pattern of the sweat pores on the finger, and detecting the fingerprint's perspiration properties (using methods different than dynamic). Next, we discuss these techniques in the context of mobile biometrics.

Hardware-based techniques: The integration of hardware-based measures into mobile devices can be challenging and is subject to cost and physical space constraints. We briefly examine some of these technologies developed by Apple and manufacturers of the various Android-based devices.

Optical fingerprint scanners are the oldest method of capturing and comparing fingerprints that rely on capturing an optical image and using algorithms to detect a user's biometric patterns, such as ridges and valleys (see Fig. 16.2), by analyzing the lightest and darkest areas of the optical image. The major drawback of optical scanners is that they are not difficult to bypass, since only 2-D pictures are captured

Fig. 16.2 Fingerprint ridges and valleys



and can be replaced with prosthetics or other high-quality pictures. Therefore, this technique is not widely used in modern devices.

The most commonly found type of modern fingerprint scanner is the capacitive scanner. Such a scanner was used in iPhone 5s, which was the first mobile device produced by Apple to support fingerprint recognition. The fingerprint reader used a capacitive sensor to read the pattern of fingerprint ridges and valleys (see Fig. 16.2). Rather than creating an optical image of a fingerprint, capacitive fingerprint scanners use arrays of tiny capacitor circuits to collect data from a user's fingerprint. The advantage of such sensors compared to traditional optical sensors, which simply take photographs of the ridges and valleys, is that capacitive sensors actually require that the finger applied to the sensor has the proper shape. Therefore, such sensors cannot be deceived by simple attacks wherein the attacker applies a fingerprint image to the sensor. Capacitors store electrical charges that are connected to conductive plates on the surface of the scanner to track a fingerprint's details. The charge stored in the capacitor will be changed slightly when a finger's ridge is placed over the conductive plates, and the air gaps between ridges will leave the charge at the capacitor unchanged. An op-amp integrator circuit is used to track these changes by causing the output to respond to changes in the input voltage over time. The result is then recorded by an analog-to-digital converter.

The latest fingerprint technology is an ultrasonic sensor, and Qualcomm's [38] Sense ID [39] ultrasonic fingerprint sensing technology is a major player in this arena. In order to capture the biometric details of a fingerprint, the hardware is composed of both an ultrasonic transmitter and a receiver. An ultrasonic transmitter transmits a pulse against the finger that is placed over the scanner. Some pulses are absorbed, while others are bounced back toward the receiver—depending on the type of biometric traits, such as ridges, valleys, and pores. Hence, depending on the signals received, a map of the fingerprint features is created. These types of scanners require that the fingerprint to have proper shape and hence cannot be deceived with a simple fingerprint photograph.

In order to prevent fingerprint spoofing, anti-spoofing technology can be implemented in software, hardware, or both. Hardware-based solutions have the advantage of a greater ability to detect the liveness of the finger that is scanned, but require additional hardware capabilities in the fingerprint scanner—such as the ability to sense pulse, temperature, and capacitance—that cannot be performed using software alone.

Typical fingerprint anti-spoofing systems measure parameters such as temperature, electrical conductivity, pulse oximetry, and skin resistance, and the built-in logic ensures the sensed value is within an acceptable range. The system includes a fingerprint sensor to capture fingerprint image data, coupled with a spoof detection module that may consist of the following components:

1. Logic that is programmed to determine the probability of a spoof from a combination of metrics derived from the fingerprint image data.
2. A metric generator is included to generate the metrics, and classifier logic is included to generate the raw probability from the metrics that the fingerprint image data was generated from a synthetic material.

3. Adjustor logic is included to adjust the raw probability by a base probability to generate the spoof probability. The base probability is generated from stored metrics based on fingerprint image data captured during an enrollment step.
4. A filter is included to divide the fingerprint image data into multiple windows. The classifier logic is also programmed to determine the spoof probability based on a comparison of the values computed from each of the windows.
5. An access module is coupled to a host system that is programmed to grant access to the host system when the spoof probability is within a predetermined range. This host system can be any one of various electronic devices, such as a smartphone, touchpad, digital camera, personal computer.
6. A storage is also included to store the metrics that are obtained in the previous steps. The storage is coupled to the spoof detection module over a network, and the stored metrics are encrypted.
7. A metric calculator is included and coupled to the classifier logic. The metric calculator is programmed to calculate multiple metrics from fingerprint image data.

Software-based techniques: Next, we discuss the dynamic and static software-based techniques that analyze sequences of fingerprint images to detect spoofs. Image sequences can be captured while the user holds the finger to the sensor for a few seconds. Marasco and Ross documented the following dynamic techniques:

1. **Perspiration-based techniques**: It is common for fingers to perspire. These approaches analyze a sequence of fingerprint images captured over a short period of time to track the progressive flow of sweat that originates in the sweat pores (located along the fingerprint ridges) and moves across those ridges. The presence of sweat makes the ridge areas between pores appear darker than the surrounding areas. The presence of these patterns is evidence of a live finger. To the best of our knowledge, this approach has not yet been attempted on mobile devices although presents an interesting research opportunity.
2. **Ridge distortion techniques**: When the finger is moved around the sensor while being pressed, the resulting fingerprint image becomes distorted. Unique properties of the skin produce significantly greater distortion for a live finger than a spoofed one. The amount of distortion can be measured by assuming the first image is non-distorted and then comparing the distortion in the first image to the other images in the sequence. Specifically, the system can look for a positive correlation between the increase of the fingerprint area and the intensity of the signal, both of which occur when pressure is applied to the surface of the finger.

Static features: Techniques in the category rely on a single image rather than a sequence, which makes the approach more efficient albeit less robust. Static features include the unique texture of the skin, properties of the skin elasticity, or perspiration-based features.

Live and spoofed fingerprints have different textures characterized by morphology, smoothness, and orientation. Marasco and Ross identified the following texture-based approaches:

1. **Texture-based**: Materials such as silicon and gelatin that are commonly used for creating spoofed fingerprints tend to be less smooth than the skin of a real finger. The extra coarseness can be measured in terms of the standard deviation of the remaining residual noise after the original image is denoised, wherein a larger deviation would be associated with a coarser surface. This approach, as is the case with most static approaches, is expected to scale favorably to mobile devices. Avila et al. [40] seem to agree, and they discussed this technique in their technical report on state-of-the-art liveness detection measures for mobile devices.
2. **First- and second-order statistics**: A live fingerprint can be distinguished from the spoof based on the differences between probabilities in observing a particular gray value at the random location on the image or based on the non-uniformity of gray areas distributed along the ridges due to sweat pores and other factors in the fingerprint anatomy. First-order statistics (i.e., mean, energy, entropy, median, variance, skewness, kurtosis, and coefficient of variation) can be used to model the distribution of gray levels, while second-order statistics construct the joint gray-level function between pairs of pixels. Both types of statistics can be efficiently computed on a modern mobile device.
3. **Local-ridge frequency analysis**: This approach [41] is based on multi-resolution texture analysis and inter-ridge frequency analysis. It measures how the distribution of the gray levels in the fingerprint image changes in response to the changes in the fingerprint structure. Moreover, *cluster shade* and *cluster prominence* features are used, both of which are computed based on the co-occurrence matrix constituting the joint probability function of two elements in a given direction and distance. Finally, these multi-resolution analysis features are combined with ridge frequency features, and a fuzzy-C-means classifier is then used to classify the combined feature set as legitimate or illegitimate.

As Marasco and Ross point out, this approach benefits by *not* depending on the perspiration phenomenon. However, local-ridge frequency analysis can be affected by cold weather, skin conditions, and dirt and moisture on the finger. This can be very problematic in mobile use cases wherein fingerprints are expected to operate in uncontrolled conditions that often include the aforementioned situations.

4. **Local phase quantization (LPQ) analysis**: A fingerprint can be rotated in many different ways. A rotation invariant LPQ technique can identify the spectral differences between a legitimate fingerprint and a spoof. The technique has the advantage of remaining robust against blurring and is likely to scale well to the mobile device's limited resources, as evidenced in the work by Jiao and Deng [42], who used LPQ in an indoor positioning application based on the mobile device camera.
5. **Power spectrum analysis** [43]: Creating a spoofed fingerprint changes the frequency details between the ridges and valleys of a fingerprint. This, in turn, results in a spoofed fingerprint image containing fewer high-frequency characteristics than a live fingerprint. The amount of high-frequency data can be computed using Fourier transform. There are currently many libraries, such as TarsosDSP [44],

that support Fourier transform. Therefore, we believe the technique is likely to prove viable for mobile devices.

6. **Local binary patterns (LBPs)**: Real and spoofed fingerprint images have different textural characteristics that can be described using LBP features. The original LBP algorithm was proposed by Ojala et al. [45] and assumes a localized 3 × 3 pixel image region. LBP is then computed by subtracting the gray value of the center pixel from the other pixels in the region. If the difference is less than or equal to 0, then the result is 0; otherwise, it is 1. Thus, the values of the surrounding pixels are binarized. Finally, the binarized value of each pixel is multiplied by the original value, and the results are summed in order to obtain the LBP operator. Mobile device libraries, such as OpenCV [46], include functions for extracting LBP.

Nikam and Agarwal [47] fused LBP features with wavelet-based features to represent ridge frequency and orientation information. The dimensionality of the fused dataset was then reduced using the Sequential Forward Floating Selection (SFFS) algorithm [21] and was then classified using a hybrid classifier approach that combined neural networks, a support vector machine, and the *k*-nearest neighbor (*k*-NN).

Jia et al. [48] argued that the 3 × 3 area fails to capture some useful textural information of the fingerprint. To address this, they proposed a multi-scale LBP operator (MSLBP). In their work, they utilized two approaches: (1) increasing the radius of the area beyond a single pixel, and then (2) applying filters to the original image as well as applying an LBP operator in the fixed radius. Evaluation on the Liveness Detection Competition 2011 (LivDet2011) database [49] showed a significantly greater increase in spoofing detection accuracy compared to the traditional LBP approach.

A more recent work by Kumpituck et al. [50] proposed that LBP be used to characterize the local appearance of sub-band images— images coded using the sub-band coding technique that decomposed the image into different constituent frequencies and then encoded each frequency separately. They first decompose the original image using a two-dimensional discrete wavelet transform (2D-DWT) in order to obtain a sub-band image. LBP extraction is then performed on the resulting image, and the extracted features are used to train the SVM classifier. The trained classifier is then used to classify images as either live or spoofed. The authors then evaluated their approach using the LivDet [49] database containing spoofed fingerprint samples and reported that using LBP derived from the sub-band images more significantly improves spoofing classification accuracy compared to the traditional approaches that use wavelet energy from sub-band energy. 2-DWT [51] as well as LBP extraction has been previously performed on mobile devices and is thus computationally viable.

Other approaches: Other static approaches include Weber Local Descriptor [52] and Binarized Statistical Image Features [53], both of whose computational demands consist primarily of linear algebra operations that can be efficiently implemented on modern mobile devices.

Perspiration-based features:

1. **Individual pore spacing**: Perspiration around the perspiration pores results in a recognizable pattern of gray levels. FFT can be used to detect these patterns. FFT is currently well supported through existing developer libraries for mobile devices [54].
2. **Intensity-based features**: Works in this category attempt to distinguish between real and spoofed images based on the uniformity of gray pixel distribution. Researchers have observed that live fingerprints have a non-uniform distribution of gray levels as well as high ridge/valley contrast values. In addition, depending on the material used to create the spoofed fingerprint, the spoofed fingerprint images have been observed as exhibiting less variation in the gray levels.

The conversion of gray images into grayscale and the analysis of pixel values comprise a computationally non-intense process performed routinely in image processing applications implemented on mobile devices.

Quality-based features: These approaches focus on discriminating between live and spoofed fingerprints based on image quality. Quality differences can be measured in terms of strength, continuity, and clarity of ridges. The hypothesis here is that spoofed images will be weaker, less continuous, and exhibit fewer clear ridges. The continuity can be measured by considering the energy concentrations, which can be computed using basic statistical and linear algebra techniques. For example, the ridge strength can be computed as a ratio of eigenvalues of the covariance matrix and the gradient vector. Similarly, the ridges' clarity can be computed using the mean and standard deviation of the foreground image [52]. The relatively low overhead of such computations makes them well suited for implementation on mobile devices.

Furthermore, as Marasco and Ross pointed out, pores located along the ridges are difficult to spoof. Therefore, integrating quality-based features into mobile device systems may be a promising approach to add yet another obstacle for frustrating fingerprint spoofing.

Pore-based approaches: Manivan et al. [55, 56] firstly used a high-pass filter to identify active sweat pores and secondly used a correlation filter to determine their position. Others have experimented with techniques to analyze the number [57] and distribution of pores [58] as well as the detection of active pores [59, 60] on the fingerprint image, the main hypothesis being that differences exist between live and spoofed images.

Rattani et al. suggested that the existing software-based anti-spoofing fingerprint methods are not robust across fingerprint fabrication materials [61]. The performance significantly drops when the fingerprint—fabricated using novel materials—is classified during the testing stage. To mitigate the impact of novel fabrication materials, automatic adaptation [62], image preprocessing [63], and open-set, classification-based [62] anti-spoofing schemes are proposed.

The above techniques should be computationally scalable to modern mobile devices. Multiple packages that support efficient implementation of low- and high-pass filtering techniques are currently available for Android [64, 65]. The remaining statistical techniques used in these approaches may either be implemented from scratch or leverage the utilities provided by existing libraries, such as TensorFlow [66]. In terms of effectiveness, more evaluation is needed for images obtained from mobile device fingerprint readers. However, similar to previous techniques, we believe the integration of these techniques provides a promising means to add yet another obstacle to frustrating spoofing attacks.

16.3.4 Voice Presentation Attacks

Voice is an appealing modality for use with mobile devices, as it allows users to interact with the device naturally through speech—the most common means for human communication. It is currently being used with Android and iPhone devices to interact with digital agent programs, thus allowing the user to perform tasks on the device by iterating commands. In addition, Android's voice unlock feature allows users to unlock their devices by uttering "Ok Google." The feature recognizes users based on their unique voice characteristics.

However, the use of voice recognition for secure authentication on mobile devices remains limited. We believe this is a result of the difficulty associated with the threat of voice spoofing attacks. Indeed, Google's support warns users of voice unlock: "You can let 'Ok Google' unlock your device when your Google Assistant recognizes your voice. Note: This setting can make your device less secure. A similar voice or recording of your own voice could unlock your device" [67]. This warning refers to the well-documented threat of a replay attack where the impostor records the legitimate user's voice and then replays it. Young et al. [68] have analyzed replay attack vulnerabilities in mobile device voice recognition systems and have proposed replay attack methodologies that can be performed using easily available software and hardware (e.g., the Raspberry Pi computing device [69]). They built a device that connects to the victim's phone and injects commands to the phone's digital assistant. Google's warning also refers to attacks involving zero-effort impostors, wherein the impostor simply speaks in his/her original voice hoping the system will mistake his/her voice for that of the legitimate user, or more sophisticated attacks wherein the impostor uses electronically synthesized speech or attempts to speak in a way that mimics the speech of the target user (such an impostor may potentially require significant training and experience).

Voice recognition can either be text dependent, wherein the same phrases must be used during the enrollment and authentication stages, or text independent, wherein any phrase can be uttered during authentication and the recognition is based on the sound of the user's voice. Text-dependent recognition is generally associated with achieving greater recognition accuracy with shorter phrases [70] and hence proves more convenient for mobile devices than text-independent speech recognition. Both types of systems, however, would face the challenge of dealing with spoofing attacks. The effort involved in simply recording and replaying a voice can be as simple as using an application on another mobile device to record a legitimate user while he/she attempts to unlock his/her device. Therefore, any user with a mobile device can be a potential attacker, which includes 77% of Americans as of 2018 [71]. We discuss potential solutions from traditional, non-mobile voice recognition systems that may prove useful in the context of mobile biometrics.

Spoofing vulnerabilities: Commonly used voice features include short-term spectral, prosodic, or high-level features. Short-term spectral features are derived from short voice frames (e.g., 20–30 ms long) and are used to describe voice timbre. Commonly used short-term spectral features include mel-frequency cepstral coefficients (MFCCs), linear predictive cepstral coefficients (LPCCs), and perceptual prediction (PLP) features [70].

Prosodic features are syllables and words that describe speaking style and intonation. The use of prosodic features for authentication may not be ideal with mobile devices because they require relatively significant training data, which might be inconvenient for the user to supply. In addition, prosodic features based on pitch are not robust in uncontrolled conditions [70] in which mobile devices operate.

High-level features include word usage, pronunciation, and other types of information that can be parsed from discrete tokens of speech. These can be robust to environmental noise but may require preprocessing in order to convert speech to text from which high-level feature extraction is possible.

All three types of features can be spoofed. Short-term spectral features can be spoofed by simply recording and replaying speech. Modern voice synthesizers are also capable of reproducing short-term spectral features if given the model of the speaker's voice.

Prosodic features can also be reproduced using synthesizers and voice conversation systems. One approach is to use a voice synthesizer to generate fundamental frequency trajectories that are correlated with the voice of the speaker being impersonated [70].

High-level features are based on speech content and can thus be easily spoofed by replaying the speech, which will have the same spoken phrases as the original voice. Moreover, artificial intelligence systems and statistical models can be used to generate speech with content sufficiently similar to that of the impersonated speaker. Next, we discuss specific threats and their countermeasures. Specifically, we discuss countermeasures to attacks based on recorded and replayed speech, synthetic speech, and voice conversion and impersonation. Our discussion is guided by the survey published by Wu et al. [70] although relates the attacks and countermeasures to mobile use cases and presents discussions of modern publications specifically targeted toward mobile devices.

Record and replay attack countermeasures: The original approach to detecting recorded and replayed speech was proposed by Shang and Stevenson [72]. The approach is based on storing voice samples from past authentication attempts and comparing these samples to the access phrase used during the authentication attempt. The attack is considered a replay if the new sample closely matches one of the prior samples. Such a technique may prove impractical for use with mobile devices, as storing all prior samples would likely result in excessive storage space consumption. In addition, the attacker might be able to obtain a sample recording sample of the user (e.g., from an online video) that was not previously used for authentication.

Villalba et al. proposed that the increased noise and reverberation resulting from replaying far-field recordings be used to detect spoofing [73]. Although the technique was effective in significantly reducing the false acceptance rate (FAR), it was attempted on both the landline and GSM telephony systems yet not on mobile devices. The approach's effectiveness for modern mobile devices remains unclear because much depends on the microphone and speaker technologies used in the attacks; these can also vary widely across devices.

Wang et al. used channel noise to detect voice samples that were recorded and replayed. The hypothesis was that the voice sample originally recorded from a live human being would only contain channel noise from the device used by the voice recognition system [70, 74]. A sample obtained from a replayed recording would also contain channel noise from the recording device and the speakers used for replay. The approach was effective in reducing equal error rates (EER) from 40.17 to 10.26% when a system based on Gaussian mixture model–universal background model (GMM-UBM) was subject to spoofing attacks. We believe this technique can scale to the limited computational resources of mobile devices, as GMMs have previously been used in mobile speech applications [75]. The technique's effectiveness in practice would require evaluation using a database of voice samples recorded on a mobile device containing spoofed samples.

Synthetic speech attack countermeasures: Many techniques have been proposed for countering attacks involving synthesized speech. These efforts are in good measure considering that the vulnerability of voice recognition systems to voice synthesis attacks is a well-recognized problem (e.g., [67]).

The synthesis processes are known to introduce detectable artifacts. Satoh et al. have used intra-frame differences that were later demonstrated to work well for synthesizers based on hidden Markov models (HMMs) that do not employ global variance compensation [70]. Other artifacts have been observed, such as the smoothing of high-order cepstral coefficients by the HMM training and synthesis processes resulting in synthetic speech containing less variation than speech originating from a living human being [70, 76]; furthermore, some researchers have focused on studying the acoustic differences between natural and synthetic speech as a means of detecting spoofing attempts. Although the above approaches may scale nicely to modern mobile devices, to the best of our knowledge, few works have focused on countering the threat of speech synthesis attacks on mobile devices.

Voice conversion attack countermeasures: While speech synthesis attacks convert text to speech, voice conversion attacks use speech samples from the targeted user to automatically convert an impostor's voice into a voice that sounds similar to that of the target speaker [70]. The conversion process introduces detectable artifacts that include the absence of the natural phase in converted speech [77, 78] and more decreased dynamic variability compared to natural speech [79]. The authors in [79] also demonstrated that supervector-based SVM classifiers can effectively detect voice conversion attacks based on utterance-level and dynamic speech variability [80], while the approaches based on detecting natural speech phases were argued to likely prove ineffective for cases wherein converted speech preserved the natural phase feature [81].

SVM-based machine learning has been widely used with mobile devices, including by the authors, which scaled adequately [82]. Therefore, we believe the implementation and evaluation of the technique in mobile applications is a viable topic for further investigation.

Human-based voice impersonation attack countermeasures: In contrast to the speech synthesis or speech conversion attacks that involve the use of technology to impersonate the voice of another person [70], a human-based voice impersonation attack does not require any additional equipment, but rather simply involves one person attempting to speak in a way that resembles another. The studies evaluating the effectiveness of these types of attacks have reported contradictory findings and are thus inconclusive.

Part of the challenge in developing countermeasures against this type of attack is that human-based voice impersonation involves the use of natural speech and hence often lacks the detectable artifacts resulting from record and replay, voice synthesis, and voice conversion [70]. Nevertheless, Chen et al. [83] successfully used the Spear system developed by Khoury et al. [84] in order to construct a mobile system resilient to human-based spoofing attacks. The system was implemented on Android 4.4 KitKat smartphone and was based on the Gaussian mixture and intersession variability (ISV) techniques. The system yielded low FARs when evaluated by the Carnegie Mellon University (CMU) Arctic Database [85].

Other recent countermeasures: Chen et al. [83] proposed a software-based approach for mobile devices that detect recorded and replayed voices based on the magnetic field emitted by the speakers. The hypothesis is that, unlike humans, loudspeakers use magnetic force to create sound that in turn produces a magnetic field that can be detected using the magnetometer sensor within a mobile device. The authors also used a Spear system [84], as described in the previous section, to detect human impersonation attacks. The overall system was able to achieve 100% accuracy.

Feng et al. [86] developed a small wearable device that protects mobile device digital assistants against replay, speech synthesis, and human-based impersonation attacks. The device includes an accelerometer that is agitated by the speech signal. The accelerometer data is then communicated via Bluetooth to the mobile device, where it is correlated with the sound data received from the mobile device microphone. This correlation is then used to perform matching on the remote server. The system produced a 0.1% false positive (or acceptance) rate. Although the wearable component may present usability concerns for users, it also presents new opportunities. For example, the wearable component can also take on the function of the security token used in multifactor authentication. Matching the voice on the remote server may prompt privacy concerns from users who fear their voices might be recorded and stored on remote systems for espionage purposes.

Zhang et al. [87] developed a mobile-based approach using the Doppler phenomenon to resist replay and human-based spoofing attacks. When the user utters a passphrase, the phone's speaker emits a 20 kHz tone—a high-frequency sound inaudible to the human ear—and monitors the microphone to pick up the signal reflections. Those resulting from the movements of the user's lips, vocal chords, etc.

while uttering a passphrase cause Doppler shifts that are used to evaluate the voice's liveness. During evaluation, the system achieved a 1% error.

Zhang et al. [88] proposed a voice replay attack detection system that leverages mobile devices' stereo sound recording capabilities. The central idea is that a stereo recording system uses two microphones, and when the live user speaks while holding the phone close to his/her mouth, the voice signal arrives at the two microphones at different times. The same phenomenon does not manifest in the case of replayed recordings.

16.3.5 Multimodal Biometrics

Using multiple biometrics requires the user to provide more evidence in order to prove identity and hence increase the amount of identifying data the attacker needs to spoof. However, combining data from multiple biometrics in a way that makes the system resilient to spoofing attacks is challenging.

Rodrigues et al. [89] empirically demonstrated that, in multimodal biometric systems that combine match scores from different modalities (using weighed sum, likelihood ratio, and Bayesian likelihood ratio), bypassing a single modality may suffice to bypass the entire system. Therefore, the multimodality of such a system simply presents the attacker with opportunities for spoofing.

Combining identifying data at the feature level is associated with greater recognition accuracy compared to combining match scores. We have previously developed feature-level fusion schemes for mobile devices that achieved significantly lower errors [82, 90, 91] compared to unimodal schemes in the presence of zero-effort impostors. We also believe feature-level fusion is a more promising approach toward multimodal systems' resilience to spoofing attacks than are methods based on score-level fusion of modalities. We are currently in the process of evaluating the performance of these schemes against spoofing attacks.

Below, we first propose methods and techniques for strengthening multimodal biometric systems on mobile devices against spoofing attacks by dividing the approaches into software- and hardware-based.

Software-based: We believe that, as the first line of defense, a multimodal system on a mobile device must perform spoofing detection on the individual modalities. Ideally, at the software level, the spoofing detection on each modality should be performed using multiple techniques to maximize the probability of detection.

The second line of defense may constitute techniques that exploit the system's multimodality to detect spoofing. For example, within a system based on face and voice, the voice signal can be correlated to the movement of the lips. A system based on the face and ears also presents multiple opportunities for increased spoofing detection. For example, our research group is currently researching the feature-level fusion of face and ear biometrics on a mobile device. The user interacts with the system by looking straight at the camera, which captures the face, and quickly turns his/her head to both the left and right such that the camera may capture both ears.

We believe the properties of these motions can be analyzed to detect replay attacks by, for example, analyzing the subtle sound made by the motion or using the device accelerometer to measure vibrations created by the motion.

The third approach to detect spoofing of a multimodal biometric system involves studying the properties of a fused set of features where one or more modality is being spoofed. We believe a correlation of the fused feature's properties set to modality spoofing may present a new line of promising research. Such research is currently being undertaken by our research group.

Finally, the software-based techniques should also be backed up by sensor-level techniques and other hardware-based techniques that may further increase the resilience of the system.

Hardware-based: To have a multimodal biometric system that is reliable and effective, it is necessary that embedded hardware be employed—e.g., low-power processor, digital signal processor (DSP), or field-programmable gate array (FPGA). These hardware resources can be used in various real-life applications, such as the authentication of electronic identification for driver licenses and e-passports, user authentication within financial institutions, and entry control within buildings, laboratories, and borders. As stated above, multimodal biometric systems can raise security to another level by adopting more than one biometric trait.

Most multimodal biometric systems are required to have powerful computing environments in which complex tasks can be executed at reasonably high speeds. Using software alone, it is not easy to process multiple biometric traits with different features in a reasonable amount of time. Therefore, we need a multimodal biometric system with support from efficient hardware in which various multimodal biometric algorithms are performed on a real-time basis. Typical application processors used in most embedded systems work at a clock rate of only a few hundreds of MHz, and the floating-point arithmetic is not hardware-implemented. However, multimodal biometric algorithms that process multiple biometric traits in parallel require higher computing power with hardware-implemented floating-point arithmetic in order to ensure a real-time authentication.

In order to perform multimodal biometrics in real time, some tasks and executions that require high computational power can be implemented into FPGA. These tasks are dynamically synthesized on FPGA, and the multimodal biometric algorithms can be processed significantly faster. Most multimodal biometric algorithms are directly related to digital image processing because multiple biometric traits, such as faces and fingerprints, are required. In the recent years, embedded system performance has been increased due to the development of new hardware such as low-power processors, DSP chips, and FPGAs. Among the hardware, FPGA is a promising technique to be used in multimodal biometric systems because it may accelerate the execution of algorithms and offer tremendous potential toward improving overall performance through parallelization [92].

Although recent new processors' technology continuously improves the performance of multimodal biometrics, the potential of implementing these algorithms on the CPU is still not fully exploited. An FPGA device can accelerate the execution of algorithms and offer a tremendous throughput by employing parallelization. On the other hand, for multimodal biometrics, the FPGA cannot accommodate all required algorithms. Therefore, optimization at the software level and hardware implementation at the hardware level must be carefully considered. Herein, the software and hardware co-design is used to design the system, which consists of both a hardware platform and software platform. The hardware employs Intel DE5 board within two DDR3 SODIMM slots that can be used to expand the amount of memory available to the FPGA. The FPGA board connects with CPU through peripheral component interconnect express (PCIe). Consequently, the biometric algorithm (e.g., face fingerprint modules) runs on the hardware platform. Our experimental results reveal that the proposed software and hardware hybrid platform can achieve three times the acceleration that the software counterpart can achieve.

16.4 Template Security

The widespread use of the biometric systems requires massive storage of biometric data. In the generic biometric authentication system, there are five major components: sensor, feature extractor, template database, matcher, and decision module (see Fig. 16.3). In Fig. 16.3, two procedures of the biometric system are depicted. During the enrollment procedure, the user information is stored in the template database. On the other side, the biometric sensor is the interface between the user and the authentication procedure. The function of the biometric sensor is to collect the biometric trait of the user. Then, the quality assessment model determines whether the collected biometric trait is sufficient for further processing. The feature extractor processes the collected biometric data to extract salient information for distinguishing between different users. Once the user information can be found in the template database, the matcher module can execute a program that compares two inputs from

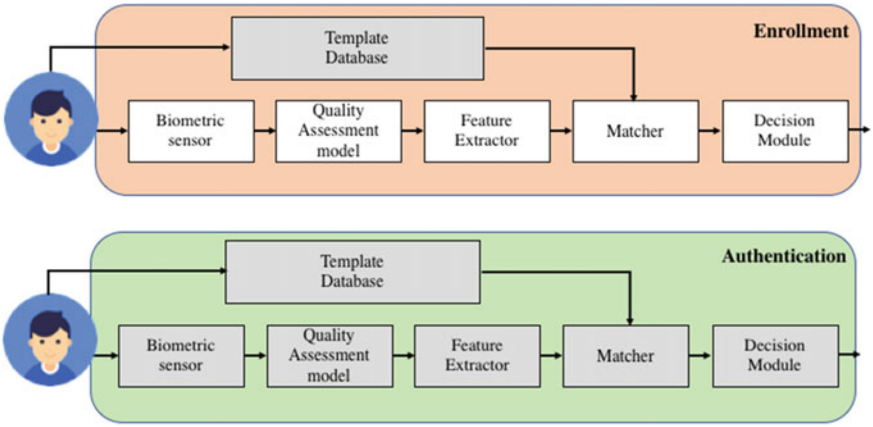


Fig. 16.3 Enrollment and authentication stages in a biometric system

the template database and feature extractor as well as generates the output as a match score. Finally, the decision module makes the decision.

The protection of the template database is not a trivial task, and some works have employed template protection schemes to improve security in a template database [93]. Non-reversibility is introduced to define the computationally infeasibility of recovering the unprotected template from the protected template. Therefore, individuals exploit the possibility of creating different protected templates from the same template used in various applications—a property known as *diversity*. Consequently, diversity leads to *revocability,* which involves protecting as many templates as necessary. Current template protection schemes can be divided into two categories: *feature transformation systems* and *biometric cryptosystems*. Some previous works have been proposed as being inspired by feature transformation. For example, the Biohashing system combines the password provided by the user with biometric data [94]. However, this method requires many passwords to protect templates, and these passwords must be stored privately. On the other hand, biometric cryptosystems try to generate additional information for unprotected templates. The major contribution of this method is that additional data is not required to be kept private. Some works provide insight into possible attacks within the generic biometric system (see Fig. 16.1). Although the software-based solution aims to protect the template database, the delay and security of the protection module are considered major drawbacks.

Some works [95] propose a fingerprint biometric cryptosystem for an FPGA device. The results imply that accuracy is improved and delay is reduced. To implement a fingerprint biometric cryptosystem in the FPGA device, both the algorithm and hardware architecture must be considered carefully. In the algorithm aspect, biometric cryptosystems are based on the fuzzy commitment that constitutes error correction and cryptosystems techniques. The error correction code can be processed by two different types either bit-by-bit or block-by-block. Both types are applied in the biometric cryptosystems with Bose–Chaudhuri–Hocquenghem (BCH) and Reed—Solomon codes [96]. On the other hand, the cryptosystems are designated based on *QFingerMaps*, and the additional cryptosystem information is generated by a fuzzy commitment scheme, which fuses the codeword and *QFingerMap* in an obfuscated way. In this work, we employ an (Exclusive OR) XOR operator to fuse the codeword and *QFingerMap*. The main functional blocks that should be implemented on the FPGA device include *QFingerMap* extraction, the encoder to generate the codeword, the hash function to protect the generated codeword, and the decoder to correct errors.

16.5 Usability

In this section, we discuss the usability issues affecting selfie biometric systems on mobile devices. We firstly discuss the general principles of user-friendly biometric system interfaces for mobile devices, then provide insights specific to designing friendly user interfaces for mobile device multimodal systems that we have learned through our research and practice.

We then present a novel approach for performing multimodal biometrics on an FPGA that can be integrated with mobile devices and can drastically reduce execution time and power consumption in multimodal mobile biometric systems, as our results suggest. Reducing these aspects is important for improving user experience. The prototype multimodal feature-level fusion system used was taken from [82] and was based on combining face and voice biometrics using discriminant correlation analysis (DCA) as well as classification using *k*-nearest neighbors (*k*-NN). The challenge stems from implementing *k*-NN on the FPGA in a way that is viable for mobile devices.

Software-based: Any software–user interface must be designed to maximize the quality of user experience. Hence, the principles of effective interface design apply to mobile biometrics systems. In this section, we focus solely on specific challenges in mobile device authentication systems that we have learned during our research and practice. We then include a specific discussion of multimodal biometrics.

First, the biometric authentication process should be easy to enable and configure, which is especially important for users who are not technology savvy. Although mobile device manufacturers are making great strides in simplifying the process, some users do not use biometric authentication because they are unsure how to set it up (in our experience, some did not even know where to find the setting). One possible solution involves encouraging the user to utilize the device's biometric feature (if it is fit for authentication) both during and following the device setup process. It is important to ensure, however, that these encouragements be both non-intrusive and easily disabled by the user.

Furthermore, the enrollment process must minimize the amount of user effort; this includes limiting the number of training samples, providing feedback on the user's progress, and minimizing processing time. Otherwise, an initial negative experience may cause the user to give up or turn away from the feature.

Second, the biometric authentication process should be easy to invoke. Many modern devices address this by setting the authentication screen as the first image the user sees upon obtaining the device's attention—typically by pressing a button. Many fingerprint-based systems, such as those used for iPhones and Galaxy devices, allow users to authenticate immediately by placing a finger on the sensor and requiring no prior actions in order to invoke the authentication process.

A similar approach is possible with face- and voice-based biometric systems. For example, many smart home systems, such as Amazon Alexa and Google Home, allow users to get the device's attention by uttering a predefined phrase—e.g., a user can utter "Ok Google" in order for Google Home system to begin accepting commands. However, such an approach would require that the mobile device constantly monitor the microphone or camera, which in turn raises issues of privacy, false device unlocks (e.g., the camera accidentally catches the user's face), and increased power consumption. For example, Android-based phones allow users to conduct Google searches and perform other functions on their devices by uttering "Ok Google." According to previous reports, some users feel apprehensive about their devices constantly "listening" to them through the microphones [97].

Third, the interface should be intuitive when interacting with users and providing them with prompt feedback in the event that matching fails. The latter is especially important for mobile devices, which operate in uncontrolled conditions that cause false rejections. For example, if the fingerprint match fails because the fingertip is wet (assuming the system can detect moisture), then the user should be instructed to wipe his/her finger; or, if the face does not match due to insufficient lighting, then the user should be instructed to increase the brightness. We believe these hints will help reduce user frustration.

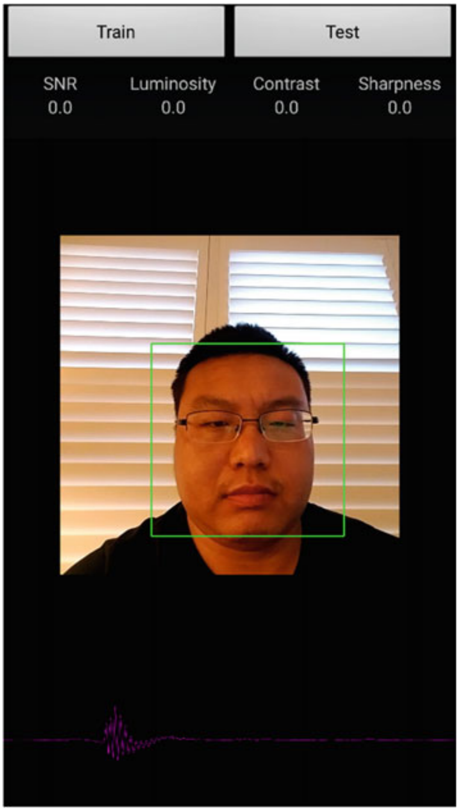
Fourth, the matching process must occur instantly, and if successful, the user should immediately be taken to the home screen of the device or to the applications he/she was most recently using. Long authentication times will likely lead to frustration, and the same is true of the enrollment process. Because the enrollment process is typically executed only once, unlike the authentication process that is done repeatedly, greater delays may be tolerated here. To maximize speed, developers can leverage the parallel architecture of modern mobile processors, graphics processing units (GPUs), and other specialized biometric technologies discussed in the section concerning hardware techniques.

Usability of mobile device multimodal biometrics: All the above user interface design principles additionally apply to multimodal biometrics. However, multimodal biometric systems require the collection of multiple biometric modalities, thus requiring greater effort from the user. We believe the key here is to minimize user efforts to a level comparable to that of a unimodal system. One possible way to achieve this is to simultaneously capture samples from multiple modalities.

For example, in our previous work, we experimented with developing an interface for a multimodal system based on face and voice (see Fig. 16.4). A user-friendly GUI for simultaneous capture of face and voice on a mobile device. The interface consists of a live stream from the device's front camera with a square drawn around the user's detected face and a volume meter indicating the strength of the voice signal. Additional indicators are provided to indicate the quality of the face (e.g., luminosity) and voice data, (e.g., signal-to-noise ratio). These indicators utilize percentages, wherein higher percentages indicate greater quality. On one hand, we believe these can help the user quickly identify issues in the event that authentication fails. On the other hand, they can potentially confuse the user with the extra data. We plan to explore the user's experienced utility of these indicators in our future research.

The system records a video of the user's face while he/she utters a phrase. The face images and voice are then extracted from the video track and sound track, respectively. The execution time for the authentication process takes a fraction of a second due to efficient algorithms and parallel extraction of both face and voice features—the most time-consuming operations of our algorithm.

We have also experimented with a multimodal biometric system based on the face and ear, finding that the easiest way for the user to capture both modalities is to look into the camera and then twist his/her head firstly to the left and secondly to the right while holding the device in a fixed position. In our informal preliminary experiments, we observed that users were able to capture both modalities within one second. Figure 16.5 presents a diagram illustrating our approach.



SNR	Luminosity	Contrast	Sharpness
0.0	0.0	0.0	0.0

Fig. 16.4 A user-friendly GUI for simultaneous capture of face and voice on a mobile device

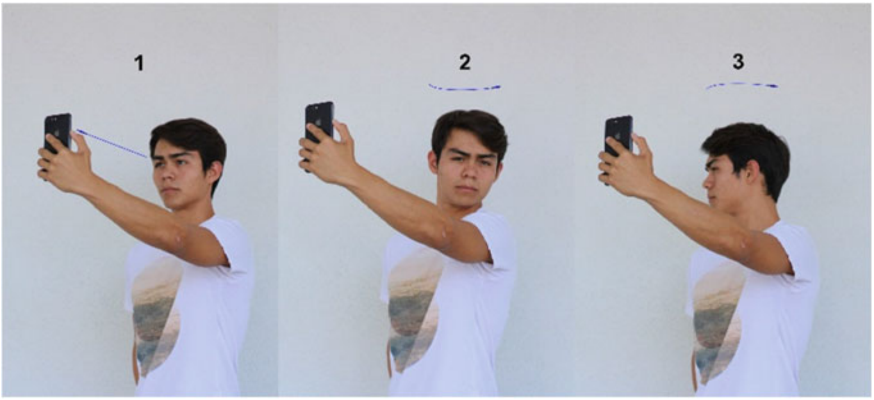


Fig. 16.5 A method for capturing face and ears in a mobile device

The above approach replaces our previous approach, which required that users move the device from the face to the left ear and then to the right ear. However, this proved to be excessively difficult for many users who could not easily find their ears with their cameras.

Samsung Galaxy S9 also includes an Intelligent Scan feature that allows the device to be unlocked with both the face and iris, which are captured simultaneously when the user looks into the camera. We believe this approach is the right direction from the interface perspective.

Expanding authentication to more than two biometrics is even more challenging; however, simultaneous capture can go a long way. For example, the effort required from a tri-modal system based on face, voice, and iris can still be achieved by recording a video of the user's face while he/she utters a phrase and simultaneously capturing images of the iris. Thus, the effort of a tri-modal system is potentially reduced to that of a bimodal system.

Overall, the research on the mobile biometrics user experience is still a relatively new field ripe for future research and innovation. It requires that designers consider technical aspects such as quick execution time, psychological aspects that involve making the system's appearance and operation inviting, and social aspects such as privacy. Achieving this goal will require that software and hardware designers as well as user experience experts join in collaboration to ensure the system is designed bottom up with usability in mind.

Next, we present a novel approach for reducing execution time and power consumption in mobile device multimodal systems using FPGAs.

Fast and power-efficient feature-level fusion of face and voice using FPGA: Current mobile devices can be used to identify users based on a single biometric modality such as the face or fingerprints. However, to attain maximum identification accuracy, prior work has revealed promising results regarding the use of multiple or multimodal biometrics.

Gofman et al. [91] proposed an approach for fusing MFCC features from the face with histogram of oriented gradient (HOG) features on mobile devices. They used DCA to fuse the features and then classified the fused feature set using various classifiers (SVM, *k*-NN, random forests, linear discriminant analysis [LDA], and quadratic discriminant analysis [QDA]). Although their approach led to more significantly improved EER compared to unimodal face and voice approaches, they did not consider the hardware aspects of implementing their approach on mobile devices (e.g., power consumption).

Recently, novel systems were proposed to incorporate programmable hardware into the smartphone to rethink a vision wherein applications may consider both software and hardware components. Current smartphone devices are incredibly constrained energy-wise due to their batteries. Development in battery density has received more attention; however, recent research shows that the battery density has been doubling only every ten years [98]. In addition, the physical size limitations of the portable devices lead to a relatively static energy budget among all devices. Consequently, CPUs empowering modern smartphones are optimized for power efficiency rather than speed. For this reason, implementing the application that requires high computation power on mobile devices nevertheless remains challenging.

k-NN was introduced as supervised and instance-based learning in the early 1950s. This algorithm was not initially popular because it requires high computation power, although it was and remains a popular means of classification in biometrics due to its simplicity and, in many cases, high accuracy. In general, there are three or four steps in the *k*-NN algorithm:

1. Calculate the distance and similarity between the testing set and the training set;
2. Sort the distance and similarity to determine the *k*-nearest classes;
3. Perform majority voting to decide the class.

There are many ways to measure the distance or similarity between data in testing and training sets. The Euclidean, Minkowsky, Chebychev, Camberra, and Manhattan methods for measuring distance are proposed as the following equations [98]:

$$\text{Euclidean: } D(x, y) = \left(\sum_{i=1}^{m} \left(|xi - yi|^2\right)^{1/2}\right)$$

$$\text{Manhattan: } D(x, y) = \sum |xi - yi|$$

$$\text{Minkowsky: } D(x, y) = \left(\sum_{i=1}^{m} |xi - yi|^r\right)^{1/r}$$

$$\text{Chebychev: } D(x, y) = \max_{i=1}^{m} |xi - yi|$$

$$\text{Camberra: } D(x, y) = \sum_{i=1}^{m} \frac{|xi - yi|}{|xi - yi|}$$

In Fig. 16.6, the hierarchical platform-based design for *k*-NN classifier is illustrated. The main purpose of the proposed design is to modularize various functions in both hardware and software. The basic operators including addition, multiplication, square root, subtraction, division, and comparator are depicted. Among various operators, *k*-NN consists of two time-consuming operations: distance computing and

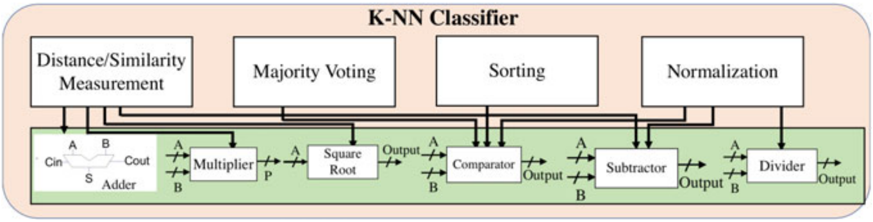


Fig. 16.6 *k*-NN classifier and its hardware functionality

sorting. Thus, these operations can be fully parallelized due to independent distance computation. The parallel property allows us to involve an FPGA device, which is perfectly suitable for implementing such k-NN heterogeneous architecture. In this work, we employ OpenCL architecture to transfer data from the CPU to FPGA. During the distance-computing process, the matrix for distance values is collected between all query and reference objects. Then, the rank process finds the k-NNs for each query object. To sort the distance, the sinking sorting algorithm is employed with a worst-case and average-case complexity $O\left(n^2\right)$. The choice of the sinking sorting algorithm in this work was based on the algorithm's property; each candidate is picked up according to the smallest distance in the current queue. The process can be perfectly parallelized because it compares each pair of adjacent items and swaps them if they are in the wrong order.

OpenCL is an open resource framework for parallel programming on systems with heterogeneous processors. Using OpenCL enables multiple hardware architectures by different manufacturers. In Figure 16.7, OpenCL framework connects host processor and FPGA through PCIe connection. The host computer handles the data flow, which is explicitly programmed by the user. The memory system in this work can be categorized into three groups: global, local, and private.

The accelerator is classified into a workgroup sharing the local memory, which plays cache-based memory such that each accelerator can access data stored in the local memory. The global memory is used to store data that is accessible to the workgroup and the host computer, while the private memory is reserved for each

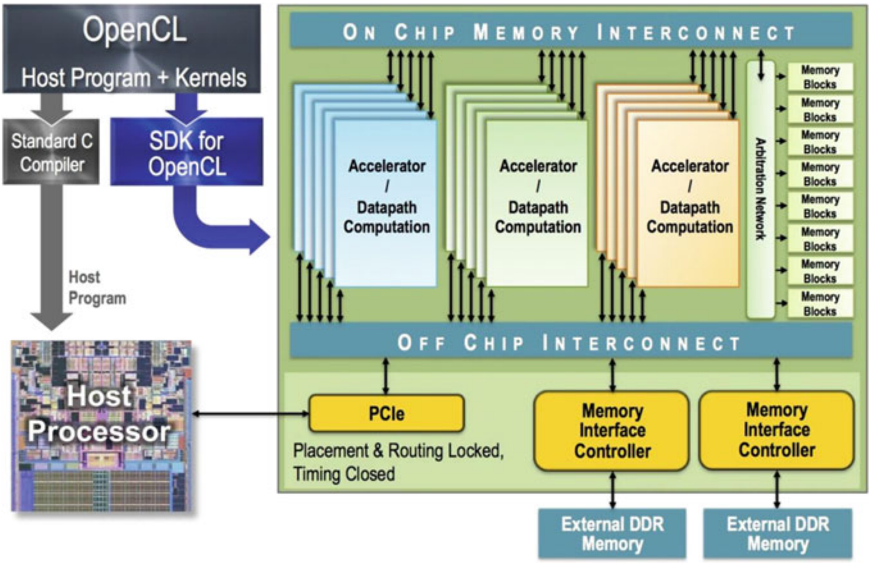


Fig. 16.7 OpenCL compiler to generate both executable software for the system CPU and a bitstream on FPGA [30,000×]

accelerator and performs the fastest data movement. The two functions are implemented in the hardware accelerator.

Compared to a traditional Verilog or VHDL design, the scheduling issue on the hardware resource is automatically attached to the device in OpenCL. Thus, in this work, we need only to design the required number of accelerators and distribute the workload.

Distance Calculation Accelerator

The design of the distance calculation accelerator aims to parallelize the distance calculation at each accelerator. In order to avoid unnecessary latency of memory use, we use local memory for distance calculation. The reference data is loaded into the local memory, which may be easily accessed by the accelerators.

Distance Sorting Accelerator

When a distance calculation accelerator produces the distance matrix between input data and reference data, the distance sorting accelerator is employed to find the *k*-nearest distance in each row using the sinking sorting algorithm. For example, when the first item compares the third and fourth distances in the row, the second item can be launched to compare others. Once all items have been compared and reached at the end of the row, the *k*-NNs are formed.

In order to test the approach, we implemented the framework in a CPU and FPGA system. The CPU used in this work was an Intel I7-3770K with 3.5 GHz with a Windows 7, 64-bit operating system. An FPGA board (Intel DE5) with a Stratix V GX was inserted and connected with CPU through PCIe lanes. The integrated transceivers with a transfer speed of 12.5 Gbps allowed the DE5 board to fully comply with version 3.0 of the PCIe standard. Two independent banks of DDR3 SODIMM RAM were used to construct global memory. The local memory employed interconnected on-chip RAM block—a simple process easily given access. Private memory was implemented using flip-flops. The flip-flop within the data flow can run at the accelerator's frequency. To test the framework, we used a labeled face and voice images from [91]. This approach's performance is compared with its CPU counterpart.

Table 16.1 illustrates the performance comparison between the CPU and FPGA results. We utilized twenty different faces and voices in order to avoid some errors during the test. Because the runtime of CPU was longer than FPGA due to unparalleled process architecture, the FPGA could thus achieve 148 times the speed up. Regarding the power aspect, CPU consumed five times that of the FPGA device.

Table 16.1 Performance comparison between CPU and FPGA

Platform	CPU	FPGA
Transistor size/nm	22	40
Runtime/ratio	150.16	1
Speedup/ratio	1	148
Power/ratio	5.41	1

16.6 Selfie Biometrics Case Studies

This section presents a couple case studies on the topic of privacy, confidentiality, and usability of selfie biometrics on mobile devices. The first relates to the face- and fingerprint-based biometric capabilities on the latest smartphones, such as Samsung (Android-based) and iPhone, while the second relates to applications of keystroke dynamics on mobile devices.

Case Study 1

The common modern smartphones mentioned above are equipped with both facial recognition and fingerprint recognition techniques. Android introduced *face unlock* in 2011 [99], while Apple introduced *Touch ID* a couple of years later [100]. These were followed by the *fingerprint lock* on the Samsung Galaxy S7 phones in 2016 and then the more recent *Face ID* technology integrated with iPhone X in 2017 [101]. This was many users' initial exposure to and true interaction with biometrics. It is thus important to assess whether users did or did not choose to adopt any of these biometric-based authentication methods to unlock their mobile devices as well as the underlying reasons behind their decisions. In fact, researchers have stated that the usability of biometric systems is a critical element in users' adoption decisions [102]. Despite the awareness of additional security provided by biometrics through passwords and PINs, people may have concerns about several issues (i.e., privacy and reliability) that act as barriers to the large-scale adoption of this technology among consumers. Hence, studies have been conducted [102–104] to explore users' beliefs, attitudes, and perceptions toward using biometric security on their mobile devices, which remains, nonetheless, not very prevalent today.

Several researchers have performed comparative studies [104, 105] to analyze usability among face recognition, iris recognition, voice recognition, fingerprint recognition, and gesture recognition techniques on mobile devices, all of which yielded considerably critical flaws. In 2014, two studies investigating smartphone unlocking behavior among users had determined that users failed to realize the importance of protecting the data stored on their phones (and hence the risks associated with losing that data) and that users spent more time than necessary to unlock their phones [106, 107]. *Face ID* on the iPhone X has recently become available and has replaced the fingerprint unlocking scheme (*Touch ID*). Reports [108] have mentioned that, although *Face ID* is perceived as more secure than *Touch ID*, there have been several issues with its operation. For example, the former does not work in landscape orientation, it does not always work in bright sunlight or with sunglasses, and it is occasionally slow.

Recently, Bhagavatula et al. [102] explored within-subject usability of *Touch ID* on iPhones and *face unlock* on Android devices in a laboratory setting in order to assess different scenarios in which mobile phones operate. Moreover, they also administered an online survey to 198 participants to evaluate general user perceptions and attitudes toward using different types of biometric security on mobile platforms during everyday life. This study was the first of its kind (at the time) to examine the usability of biometric security on commonly used smartphones in today's society.

We summarize the methodologies used and results obtained from this case study in the following section.

Laboratory usability study: The within-subject study performed by Bhagavatula et al. [102] consisted of comparing four unlocking mechanisms on smartphones: Android face unlock on a Samsung Galaxy S4 phone, iPhone *Touch ID* (fingerprint recognition), Android PIN unlock, and iPhone PIN unlock. They compared these biometric-based authentication schemes because these were the only such schemes available on smartphones at the time (Android fingerprint unlock and iPhone *Face ID* were not yet introduced). The PINs were used as a baseline for comparison among the biometric security techniques. Ten participants—eight male and two females—participated in the study, and each participant was provided with a phone. Participants also filled out a questionnaire concerning their demographic backgrounds, prior experience with smartphones and biometric systems, and perceptions and attitudes toward biometrics (Likert-scale-type questions). Each subject also performed authentication using each of the four schemes in five different scenarios: (1) sitting, (2) sitting in a dark room, (3) walking, (4) walking while carrying a bag in one hand, and (5) sitting and applying moisturizer to the hands. These five situations are consistent with prior studies of mobile phones' user usability. Their main results include

- All participants determined Android face unlock and iPhone *Touch ID* as being easy to use during several common usage scenarios;
- The face unlock did not work for any participants in the darkroom setting;
- *Touch ID* was relatively easy to use even in the presence of moisturizer on participants' hands; and
- Most participants favored iPhone's *Touch ID* over Android's face unlock and PINs.

Online survey: The purpose of the online survey was to understand real usability issues faced by consumers in the real world, such as the perceived usefulness of the biometric security schemes to protect the phone from unauthorized use and the system's ease of use or convenience. For this purpose, 198 subjects who owned a smartphone model that supported either Android face unlock or iPhone *Touch ID* were selected. Similar to the laboratory study, participants were asked through a survey to provide their demographic information, level of prior familiarity with biometric authentication techniques, general phone unlocking behaviors, and rationale for adopting or not adopting a biometric scheme for their mobile phones. The main findings from this survey include:

- Participants using iPhones overwhelmingly perceived *Touch ID* as more convenient to use than PINs, although a few users reported issues with *Touch ID* when using the phones with dirty fingers; and
- Very few Android users, on the other hand, used the face unlock technique to unlock their phones due to technical difficulties encountered.

The overall conclusions reached by Bhagavatula et al. from their two studies clearly indicate that people more positively perceive the extra security provided by biometrics on their mobile devices compared to traditional methods, such as PINs; furthermore, iPhone's *Touch ID* was determined the most popular biometric.

Android's face unlock mechanism seemed to suffer from some drawbacks that, if fixed, may lead to more large-scale adoption. In general, just as it is important to develop novel biometric authentication techniques for mobile phones, it is equally important to assess user perceptions and attitudes regarding usability in order to make mobile biometric security more prevalent among the masses (Figs. 16.8 and 16.9).

Case Study 2

The use of behavioral biometrics, such as gait and keystroke dynamics, is still not as prevalent in mobile devices as the use of the physical biometrics (e.g., face, fingerprints, and iris). However, existing research indicates that authentication methods can be improved by considering implicit, individual behavioral cues [109, 110]. Verifying identity based on typing behavior—also called "keystroke dynamics"—has been studied thoroughly in the literature with older mobile phones with physical keys [103, 111] as well as with newer devices featuring touchscreens [112, 113].

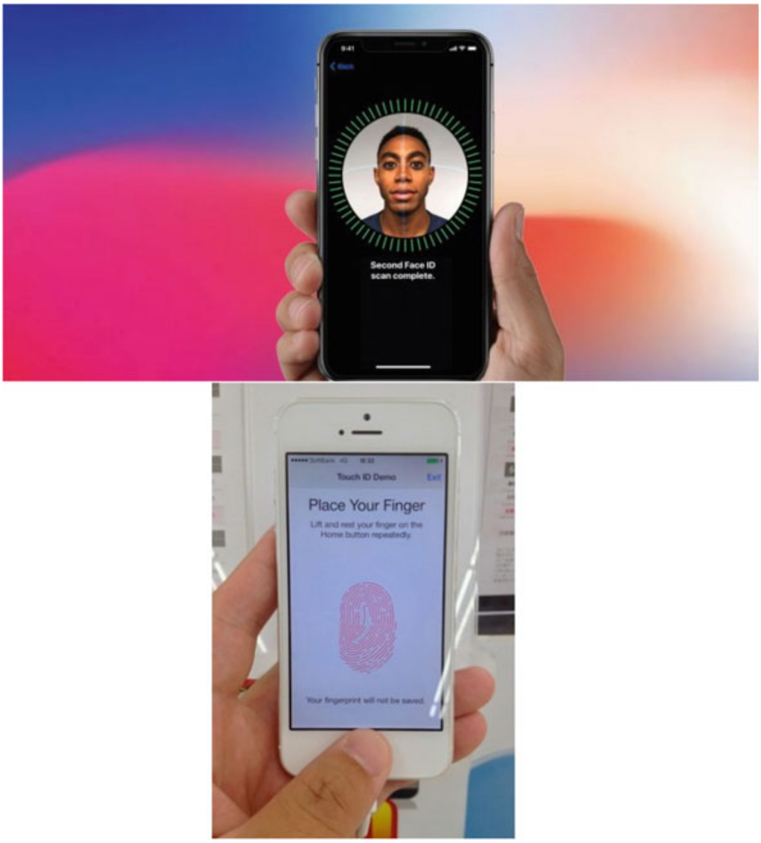

9:41
Back
Second Face ID scan complete.
Touch ID Demo
Exit
Place Your Finger
Lift and rest your finger on the Home button repeatedly.
Your fingerprint will not be saved.

Fig. 16.8 Face ID and Touch ID on Apple iPhones. *Copyright info* Google images—"labeled for reuse"

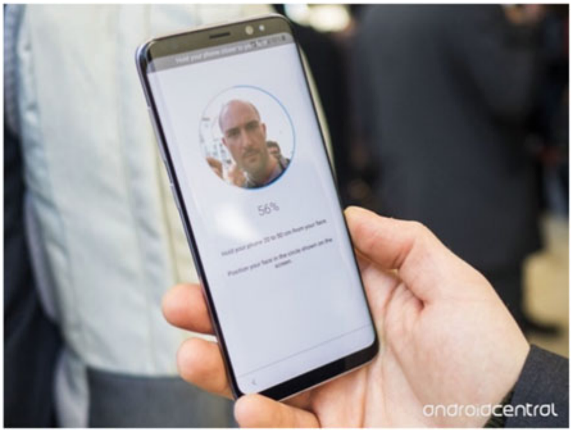


Fig. 16.9 Face unlock on a Samsung Galaxy phone. *Copyright info* Google images—“labeled for reuse”

Buschek et al. [114] presented an in-depth analysis of current keystroke biometrics on current smartphones that provide touch-typing capabilities and included a proposed approach to improve the usability of this method, which we discuss briefly as a case study in this subsection.

Buschek et al. [114] collected data from 28 participants aged an average of 25 years; eight participants were female, twenty were male, and all owned mobile phones with touchscreens and typed with their right hands. Each participant was invited to two sessions that were at least one week apart. Each session comprised three main tasks (three hand postures) and lasted about an hour. For each hand posture, participants typed six different passwords in random order twenty times each. The number of attempts was unlimited, and the user could reenter the password if a wrong attempt was entered during any step.

Some challenges for practical and usable applications of mobile keystroke biometrics are demonstrated by the study’s following results:

- The EERs obtained from data collected in a single session were lower than those collected over different sessions, indicating that mobile typing biometrics vary over time;
- Mobile typing biometrics are highly dependent on the specific hand posture; training and testing using multiple postures increased participants’ EERs by 86.3% relative to testing with the same hand posture.

These observations imply that an important consideration for improving the usability of mobile keystroke biometrics involves the ability of the application to infer postures dynamically. The latter can be achieved by combining the models for different hand postures using a probabilistic framework that has proven to reduce EERs more significantly than a single model based on one posture. Thus, although using multiple hand postures creates a trade-off between security (lower EERs) and usability, this can be easily addressed (as described above). Since usability is a primary concern for more widespread application of biometrics on the mobile platform,

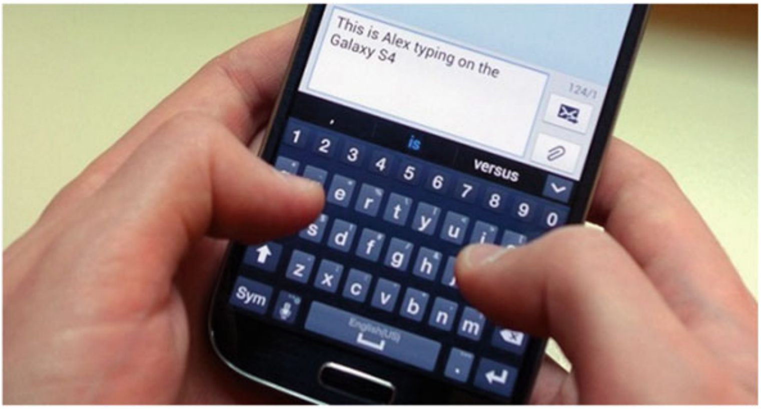


Fig. 16.10 Keystrokes on a Samsung Galaxy phone. *Copyright info* Google images—"labeled for reuse"

this case study offers interesting insights as to how this may be achieved without compromising the level of security attained (Fig. 16.10).

16.7 Conclusion

Users have been eager to embrace selfie biometrics. However, security vulnerabilities and usability issues have emerged. Researchers and mobile device manufacturers have proposed innovative software- and hardware-based techniques meant to overcome these problems, many of which yielded promising results. iPhone X's Face ID system, for example, cannot be deceived with a photograph of the person's face thanks to its imaging technology. However, vulnerabilities continue to pose a threat (e.g., 3-D masks).

As the use of selfie biometrics grows and new modalities find their way onto mobile devices, new security and usability challenges will arise and introduce ripe areas for future innovation and development.

Acknowledgements We would like to thank the students in our research laboratory for their assistance in developing and implementing the approaches discussed in this chapter—specifically, Narciso Sandico, Sadun Muhi, and Eryu Suo. We would additionally like to thank Maria Villa and Bryan Villa for their illustrative work in Fig. 16.5.

References

1. Apple Corporation (2018) IPhone 5s—technical specifications. Retrieved from https://support.apple.com/kb/sp685?locale=en_US. Cited 24 Sept 2018
2. Chaos Computer Club (CCC) (2013) Chaos Computer Club breaks Apple TouchID. Retrieved from https://www.ccc.de/en/updates/2013/ccc-breaks-apple-touchid. Cited 15 Aug 2018
3. Rogers M (2014) Why I hacked TouchID (again) and still think it's awesome. Lookout Blog. Retrieved from https://blog.lookout.com/iphone-6-touchid-hack. Cited 2 July 2018
4. Cao K, Jain AK (2016) Hacking mobile phones using 2D printed fingerprints. MSU technical report
5. Heisler Y (2017) Security researchers demo how "easy" it is to fool face ID with a 3D mask. BGR. Retrieved from https://bgr.com/2017/11/28/face-id-hack-3d-mask-iphone-x-security/. Cited 2 July 2018
6. Matteson S (2017) IPhone's face ID can be hacked, but here's why nobody needs to panic. TechRepublic. Retrieved from https://www.techrepublic.com/article/iphones-face-id-can-be-hacked-but-heres-why-nobody-needs-to-panic/. Cited 2 July 2018
7. Ratha NK, Connell JH, Bolle RM (2001) Enhancing security and privacy in biometrics-based authentication systems. IBM Systems Journal 40(3):614–634
8. Apple Corporation (2018) IOS security. Retrieved from https://www.apple.com/business/site/docs/iOS_Security_Guide.pdf. Cited 2 Sept 2018
9. Brandom R (2016) Your phone's biggest vulnerability is your fingerprint. Retrieved from https://www.theverge.com/2016/5/2/11540962/iphone-samsung-fingerprint-duplicate-hack-security. Cited 4 June 2018
10. McKane J (2018) We made fake fingerprints and hacked into a Nokia 5. MyBroadband. Retrieved from https://mybroadband.co.za/news/security/267331-we-made-fake-fingerprints-and-hacked-into-a-nokia-5.html. Cited 3 Aug 2018
11. Smith C (2017) The iPhone X's face ID has one real vulnerability: your kids. BGR. Retrieved from https://bgr.com/2017/11/14/iphone-x-face-id-hacked-children/. Cited 3 July 2018
12. Smith C (2017) Face ID shown unlocking for family members who aren't alike. BGR. Retrieved from https://bgr.com/2017/12/31/iphone-x-face-id-hack-family-members/. Cited 3 July 2018
13. Williams-Grut O (2016) A researcher claims 2 bank apps can be hacked using iPhone's "Live Photos." Business Insider. Retrieved from https://www.businessinsider.com/bank-apps-facial-recognition-hacked-using-iphone-live-photos-2016-8. Cited 3 July 2018
14. International Organization for Standardization (2016) Information technology—biometric presentation attack detection—part 1: framework. Retrieved from https://www.iso.org/standard/53227.html
15. National Institute of Standards and Technology (2013) Standards for biometric technologies. Retrieved from https://www.nist.gov/speech-testimony/standards-biometric-technologies
16. Galbally J, Marcel S, Fierrez J (2014) Image quality assessment for fake biometric detection: application to iris, fingerprint, and face recognition. IEEE Trans Image Process 23(2):710–724
17. Rattani A, Derakhshani R (2018) A survey of mobile face biometrics. Comput Electr Eng 72:39–52
18. Amadeo R (2017) Galaxy S8 face recognition already defeated with a simple picture. Ars Technica. Retrieved from https://arstechnica.com/gadgets/2017/03/video-shows-galaxy-s8-face-recognition-can-be-defeated-with-a-picture/. Cited 3 July 2018
19. Moren D (2015) Face recognition security, even with a "blink test," is easy to trick. Popular Science. Retrieved from https://www.popsci.com/its-not-hard-trick-facial-recognition-security?utm_medium=twitter&utm_source=twitterfeed. Cited 3 July 2018
20. de Freitas Pereira T, Anjos A, De Martino JM, Marcel S (2013) Can face anti spoofing countermeasures work in a real world scenario? Paper presented at the IEEE international conference on biometrics (ICB). Madrid, Spain
21. Pudil P, Novovičová J, Kittler J (1994) Floating search methods in feature selection. Pattern Recogn Lett 15(11):1119–1125

22. Erdogmus N, Marcel S (2013) Spoofing 2D face recognition systems with 3D masks. Paper presented at the 2013 international conference of the biometrics special interest group (BIOSIG), Darmstadt, Germany
23. Ahonen T, Hadid A, Pietikainen M (2006) Face description with local binary patterns: application to face recognition. IEEE Trans Pattern Anal Mach Intell 28(12):2037–2041
24. Li J, Wang Y, Tan T, Jain AK (2004) Live face detection based on the analysis of fourier spectra. Biomet Technol Hum Ident 5404:296–304
25. Cardinal D (2017) How Apple's iPhone X TrueDepth camera works. ExtremeTech. Retrieved from https://www.extremetech.com/mobile/255771-apple-iphone-x-truedepth-camera-works. Cited 14 Sept 2017
26. Apple Corporation (2017) Face ID security. Retrieved from https://www.apple.com/business/site/docs/FaceID_Security_Guide.pdf
27. InAuth (2017) Fingerprints: the most popular biometric. Retrieved from https://www.inauth.com/blog/fingerprints-popular-biometric/. Cited 2 July 2018
28. Tess (2017) Realistic 3D printed finger could make smartphone fingerprint scanners harder to hack. 3drs.org. Retrieved from https://www.3ders.org/articles/20170925-realistic-3d-printed-finger-could-make-smartphone-fingerprint-scanners-harder-to-hack.html
29. Marasco E, Ross A (2015) A survey on antispoofing schemes for fingerprint recognition systems. ACM Comput Surv (CSUR) 47(2):28
30. Bowden-Peters E, Phan RCW, Whitley JN, Parish DJ (2012) Fooling a liveness-detecting capacitive fingerprint scanner. Cryptography and security: from theory to applications. Springer, Berlin, pp 484–490
31. Phone Arena (2018) Samsung Galaxy S9 vs Samsung Galaxy S5—Phone specs comparison. Retrieved from https://www.phonearena.com/phones/compare/Samsung-Galaxy-S9,Samsung-Galaxy-S5/phones/10717,8202. Cited 3 July 2018
32. Fox-Brewster T (2018) Yes, cops are now opening iPhones with dead people's fingerprints. Forbes. Retrieved from https://www.forbes.com/sites/thomasbrewster/2018/03/22/yes-cops-are-now-opening-iphones-with-dead-peoples-fingerprints/#3e50d3c7393e. Cited 3 July 2018
33. Hardy E (2018) Cops will use Touch ID on your corpse to unlock your iPhone. Cult of Mac. Retrieved from https://www.cultofmac.com/536691/police-unlock-iphones-with-dead-fingers-touch-id/. Cited 4 July 2018
34. Wehner M (2016) Why a disembodied finger can't be used to unlock the touch ID sensor on the iPhone 5s. Engadget. Retrieved from https://www.engadget.com/2013/09/16/why-a-disembodied-finger-cant-be-used-to-unlock-the-touch-id-se/. Cited 4 July 2018
35. Etherington D (2013). Watch a cat unlock the iPhone 5s using touch ID and the fingerprint sensor. Retrieved from https://techcrunch.com/2013/09/19/watch-a-cat-unlock-the-iphone-5s-using-touch-id-and-the-fingerprint-sensor/. Cited 4 July 2018
36. Leopold T (2013) New iPhone 5S fingerprint sensor works for dogs. CNN. Retrieved from https://www.cnn.com/2013/09/20/tech/mobile/iphone-dog-paw-print-ireport/index.html. Cited 3 July 2018
37. Kooser A (2016) See a hedgehog unlock an iPhone with its tiny paw. CNET. Retrieved from https://www.cnet.com/news/hedgehog-unlock-iphone-sashimi/. Cited 4 July 2018
38. Qualcomm (2017) Qualcomm fingerprint sensors. Retrieved from https://www.qualcomm.com/solutions/mobile-computing/features/security/fingerprint-sensors. Cited 4 July 2018
39. Qualcomm (2018) Qualcomm announces advanced fingerprint scanning and authentication technology. Retrieved from https://www.qualcomm.com/news/releases/2017/06/28/qualcomm-announces-advanced-fingerprint-scanning-and-authentication. Cited 4 July 2018
40. Avila CS, Casanova JG, Ballesteros F, Garcia LRT, Gomez MFA, Sierra DS (2014) State of the art of mobile biometrics, liveness and non-coercion detection. Personalized Centralized Authentication System
41. Abhyankar A, Schuckers S (2006) Fingerprint liveness detection using local ridge frequencies and multiresolution texture analysis techniques. Paper presented at the IEEE international conference on image processing (ICIP). Atlanta, GA

42. Jiao J, Deng Z (2017) Deep combining of local phase quantization and histogram of oriented gradients for indoor positioning based on smartphone camera. Int J Distrib Sens Netw 13(1):1550147716686978
43. Coli P, Marcialis G, Roli F (2007) Power spectrum-based fingerprint vitality detection. Paper presented at the IEEE international work on automatic identification advanced technologies (AutoID). Alghero, Italy
44. Six J, Cornelis O, Leman M (2014) TarsosDSP, a real-time audio processing framework in Java. Paper presented at the audio engineering society 53rd international conference: semantic audio. London, England
45. Ojala T, Pietikäinen M, Harwood D (1996) A comparative study of texture measures with classification based on featured distributions. Pattern Recogn 29(1):51–59
46. Muhammad A (2015) OpenCV Android programming by example. Packt Publishing Ltd, Birmingham
47. Nikam SB, Agarwal S (2008) Texture and wavelet-based spoof fingerprint detection for fingerprint biometric systems. Paper presented at the 2018 first international conference on emerging trends in engineering and technology, Nagpur, Maharashtra, India
48. Jia X, Yang X, Cao K, Zang Y, Zhang N, Dai R, Tian J (2014) Multi-scale local binary pattern with filters for spoof fingerprint detection. Inf Sci 268:91–102
49. Yambay D, Ghiani L, Denti P, Marcialis GL, Roli F, Schuckers S (2012) LivDet 2011—Fingerprint liveness detection competition 2011. Paper presented at the 2012 5th IAPR international conference on biometrics (ICB), New Delhi, India
50. Kumpituck S, Li D, Kunieda H, Isshiki T (2017) Fingerprint spoof detection using wavelet based local binary pattern. Paper presented at the 8th international conference on graphic and image processing (ICGIP 2016). Bellingham, WA
51. Kumar L, Sharma K (2013) Web based novel technique for watermarking colour images on Android mobile phones. Int J Adv Res Comput Sci Softw Eng 3(7)
52. Gragnaniello D, Poggi G, Sansone C, Verdoliva L (2013) Fingerprint liveness detection based on weber local image descriptor. Paper presented at the 2013 IEEE workshop on biometric measurements and systems for security and medical applications (BIOMS). Naples, Italy
53. Kannala J, Rahtu E (2012) Bsif: binarized statistical image features. Paper presented at the 2012 21st international conference on pattern recognition (ICPR). Tsukuba, Japan
54. Superpowered (n.d.) IOS and Android FFT & iOS and Android Polar FFT. Retrieved from https://superpowered.com/fft-and-polar-fft. Cited 4 July 2018
55. Manivanan N, Memon S, Balachandran W (2010) Automatic detection of active sweat pores of fingerprint using highpass and correlation filtering. Electron Lett 46(18):1268–1269
56. Manivanan N, Memon S, Balachandran W (2010) Security breaks a sweat. Electron Lett 46(18):1241–1242
57. Espinoza M, Champod C (2011) Using the number of pores on fingerprint images to detect spoofing attacks. Paper presented at the 2011 international conference on hand-based biometrics (ICHB), Hong Kong, China
58. Marcialis GL, Roli F, Tidu A (2010) Analysis of fingerprint pores for vitality detection. Paper presented at the 2010 20th international conference on pattern recognition (ICPR). Istanbul, Turkey
59. Memon SA (2012) Novel active sweat pores based liveness detection techniques for fingerprint biometrics. Doctoral dissertation. Brunel University School of Engineering and Design Ph.D. theses
60. Memon S, Manivannan N, Balachandran W (2011) Active pore detection for liveness in fingerprint identification system. Paper presented at the 2011 19th telecommunications forum (TELFOR), Belgrade, Serbia
61. Rattani A, Scheirer WJ, Ross A (2015) Open set fingerprint spoof detection across novel fabrication materials. IEEE Trans Inf Forensics Secur 10(11):2447–2460
62. Rattani A, Ross A (2014) Automatic adaptation of fingerprint liveness detector to new spoof materials. Paper presented at the IEEE international joint conference on biometrics. Clearwater, FL

63. Rattani A, Ross A (2014a) Minimizing the impact of spoof fabrication material on fingerprint liveness detector. Paper presented at the 2014 IEEE international conference on image processing (ICIP). Paris, France
64. Bhide B (2013) Low-pass-filter-to-Android-sensors. Retrieved from https://github.com/Bhide/Low-Pass-Filter-To-Android-Sensors. Cited 4 July 2018
65. W3C Working Group (2017) Motion sensors explainer. Retrieved from https://www.w3.org/TR/motion-sensors/#low-pass-filters. Cited 4 July 2018
66. Alzantot M, Wang Y, Ren Z, Srivastava MB (2017) RSTensorFlow: GPU enabled TensorFlow for deep learning on commodity android devices. Paper presented at the 1st international workshop on deep learning for mobile systems and applications. Niagara Falls, NY
67. Google Corporation (n.d.) Change “Ok Google” settings. Retrieved from https://support.google.com/assistant/answer/7394306?hl=en. Cited 4 July 2018
68. Young PJ, Jin JH, Woo S, Lee DH (2016) BadVoice: soundless voice-control replay attack on modern smartphones. Paper presented at the 2016 eighth international conference on ubiquitous and future networks (ICUFN). Vienna, Austria
69. Richardson M, Wallace S (2012) Getting started with raspberry PI. O’Reilly Media Inc., Sebastopol
70. Wu Z, Evans N, Kinnunen T, Yamagishi J, Alegre F, Li H (2015) Spoofing and countermeasures for speaker verification: a survey. Speech Commun 66:130–153
71. Pew Research Center (2018) Demographics of mobile device ownership and adoption in the United States. Retrieved from http://www.pewinternet.org/fact-sheet/mobile/. Cited 3 July 2018
72. Shang W, Stevenson M (2010) Score normalization in playback attack detection. Paper presented at the IEEE international conference on acoustics, speech, and signal processing (ICASSP). Dallas, TX
73. Villalba J, Lleida E (2011) Detecting replay attacks from far-field recordings on speaker verification systems. European workshop on biometrics and identity management. Springer, Berlin, pp 274–285
74. Wang ZF, Wei G, He QH (2011) Channel pattern noise based playback attack detection algorithm for speaker recognition. Paper presented at the 2011 international conference on machine learning and cybernetics (ICMLC). Guilin, China
75. Rossi M, Feese S, Amft O, Braune N, Martis S, Tröster G (2013) AmbientSense: a real-time ambient sound recognition system for smartphones. Paper presented at the 2013 IEEE international conference on pervasive computing and communications workshops (PERCOM Workshops). San Diego, CA
76. Chen LW, Guo W, Dai LR (2010) Speaker verification against synthetic speech. Paper presented at the 7th international symposium on Chinese spoken language processing (ISCSLP). Tainan, Taiwan
77. Wu Z, Chng ES, Li H (2012) Detecting converted speech and natural speech for anti-spoofing attack in speaker recognition. Interspeech
78. Wu Z, Kinnunen T, Chng ES, Li H, Ambikairajah E (2012) A study on spoofing attack in state-of-the-art speaker verification: the telephone speech case. Paper presented at the 2012 Asia-Pacific signal information processing association annual summit and conference (APSIPA ASC). Hollywood, CA
79. Alegre F, Amehraye A, Evans N (2013) A one-class classification approach to generalised speaker verification spoofing countermeasures using local binary patterns. Paper presented at the international conference on biometrics: theory, applications and systems (BTAS). Alrington, VA
80. Alegre F, Amehraye A, & Evans N (2013) Spoofing countermeasures to protect automatic speaker verification from voice conversion. Paper presented at the IEEE international conference on acoustics, speech, and signal processing (ICASSP). Vancouver, BC
81. Matrouf D, Bonastre JF, Fredouille C (2006) Effect of speech transformation on impostor acceptance. Paper presented at the 2006 IEEE international conference on acoustics, speech, and signal processing (ICASSP). Toulouse, France

82. Gofman M, Sandico N, Mitra S, Suo E, Muhi S, Vu T (2018) Multimodal biometrics via discriminant correlation analysis on mobile devices. Paper presented at the 2018 international conference on security and management. Las Vegas, NV
83. Chen S, Ren K, Piao S, Wang C, Wang Q, Weng J, Su L, Mohaisen A (2017) You can hear but you cannot steal: defending against voice impersonation attacks on smartphones. Paper presented at the 2017 IEEE 37th international conference on distributed computing systems (ICDCS). Atlanta, GA
84. Khoury E, El Shafey L, Marcel S (2014) Spear: an open source toolbox for speaker recognition based on Bob. Paper presented at the 2014 IEEE international conference on acoustics, speech and signal processing (ICASSP). Florence, Italy
85. Kominek J, Black AW (2004) The CMU arctic speech databases. Paper presented at the 5th ISCA workshop on speech synthesis. Pittsburgh, PA
86. Feng H, Fawaz K, Shin KG (2017) Continuous authentication for voice assistants. Paper presented at the 23rd annual international conference on mobile computing and networking. Snowbird, UT
87. Zhang L, Tan S, Yang J (2017) Hearing your voice is not enough: an articulatory gesture based liveness detection for voice authentication. Paper presented at the 2017 ACM SIGSAC conference on computer and communications security. Dallas, TX
88. Zhang L, Tan S, Yang J, Chen Y (2016) Voicelive: a phoneme localization based liveness detection for voice authentication on smartphones. Paper presented at the 2016 ACM SIGSAC conference on computer and communications security. Vienna, Austria
89. Rodrigues RN, Ling LL, Govindaraju V (2009) Robustness of multimodal biometric fusion methods against spoof attacks. J Vis Lang Comput 20(3):169–179
90. Gofman MI, Mitra S, Smith N (2016) Hidden Markov models for feature-level fusion of biometrics on mobile devices. Paper presented at the 2016 IEEE/ACS 13th international conference of computer systems and applications (AICCSA). Agadir, Morocco
91. Gofman MI, Mitra S, Cheng THK, Smith NT (2016) Multimodal biometrics for enhanced mobile device security. Commun ACM 59(4):58–65
92. Katona M et al (2005) FPGA design and implementation of a wavelet-domain video denoising system. Lect Notes Comput Sci 3708:650–657
93. Maltoni D, Maio D, Jain AK, Prabhakar S (2009) Handbook of fingerprint recognition, 2nd edn. Springer, New York City
94. Teoh ABJ, Goh A, Ngo DCL (2006) Random multispace quantization as an analytic mechanism for BioHashing of biometric and random identity inputs. IEEE Trans Pattern Anal Mach Intell 28(12):1892–1901
95. Arjona R, Baturone I (2015) A fingerprint biometric cryptosystem in FPGA. Paper presented at the 2015 IEEE international conference on industrial technology (ICIT). Seville, Spain
96. Imamverdiyev Y, Teoh ABJ, Kim J (2013) Biometric cryptosystem based on discretized fingerprint texture descriptors. Expert Syst Appl 40(5):1888–1901
97. Pepicq B (2017) Why do some people refuse to use Google Assistant? AndroidPIT. Retrieved from https://www.androidpit.com/why-not-use-google-assistant. Cited 3 June 2018
98. Mohsin MA (2017) An FPGA-based hardware accelerator for *k*-nearest neighbor classification for machine learning. Master thesis. University of Colorado Springs
99. Paul R (2011) Unwrapping a new ice cream sandwich: Android 4.0 reviewed. Ars Technica. Retrieved from https://arstechnica.com/gadgets/2011/12/unwrapping-a-new-ice-cream-sandwich-android-40-reviewed-1/. Cited 20 June 2018
100. Apple (2018) Using touch ID on the iPhone. Retrieved from http://support.apple.com/kb/ht5883. Cited 20 June 2018
101. Chamary JV (2017) No, Apple's face ID is not a "secure Password". Forbes. Retrieved from https://www.forbes.com/sites/jvchamary/2017/09/18/security-apple-face-id-iphone-x/#99580fc4c835. Cited 28 June 2018
102. Bhagavatula C, Ur B, Iacovino K, Kywe SM, Cranor LF, Savvides M (2015) Biometric authentication on iPhone and Android: usability, perceptions, and influences on adoption. Paper presented at the usable security (USEC). Workshop. San Diego, CA

103. Clarke NL, Furnell SM (2006) Authenticating mobile phone users using keystroke analysis. Int J Inf Secur 6(1):1–14
104. Trewin S, Swart C, Koved L, Martino J, Singh K, Ben-David S (2012) Biometric authentication on a mobile device: a study of user effort, error and task disruption. Paper presented at the 28th annual computer security applications conference (ACSAC). Orlando, FL
105. Braz C, Robert J-M (2006) Security and usability: the case of the user authentication methods. Paper presented at the 18th International conference of the association francophone d'Interaction Homme-Machine. Montreal, Quebec
106. Egelman S, Jain S, Portnoff RS, Liao K, Consolvo S, Wagner D (2014) Are you ready to lock? Paper presented at the ACM SIGSAC conference on computer & communications security. Scottsdale, AZ
107. Harbach M, von Zezschwitz E, Fichtner A, De Luca A, Smith M (2014) It's a hard lock life: A field study of smartphone (un)locking behavior and risk perception. Paper presented at the symposium on usable privacy and security. Menlo Park, CA
108. Sathiah S (2017) Face ID on the iPhone X is a backwards step in usability. Notebookcheck. Retrieved from https://www.notebookcheck.net/Face-ID-on-the-iPhone-X-is-a-backwards-step-in-usability.264306.0.html. Cited 28 June 2018
109. Burgbacher U, Hinrichs K (2014) An implicit author verification system for text messages based on gesture typing biometrics. Paper presented at the ACM CHI conference on human factors in computing systems. Toronto, Canada
110. Crawford H (2010) Keystroke dynamics: characteristics and opportunities. In: 8th international conference on privacy, security and trust. Ottawa, CA
111. Campisi P, Maiorana E, Lo Bosco M, Neri A (2009) User authentication using keystroke dynamics for cellular phones. IET Sig Process 3(5):333–341
112. Nauman M, Ali T, Rauf A (2013) Using trusted computing for privacy preserving keystroke-based authentication in smartphones. Telecommun Syst 52:2149–2161
113. Saevanee H, Bhattarakosol P (2009) Authenticating user using keystroke dynamics and finger pressure. Paper presented at the 6th IEEE consumer communications and networking conference. Las Vegas, NV
114. Buschek D, De Luca A, Alt F (2015) Improving accuracy, applicability, and usability of keystroke biometrics on mobile touchscreen devices. Paper presented at the ACM CHI 2015 conference, crossings. Seoul, Korea

Chapter 17
The Horcrux Protocol: A Distributed Mobile Biometric Self-sovereign Identity Protocol

Asem Othman and John Callahan

Abstract Deployed mobile biometric authentication systems rely on mobile- or server-centric models. However, both model schemes present a single point of biometric data compromise from a security perspective. If biometric data is compromised, it poses a direct threat to users' digital identities. A recent example of compromised biometric data includes the stolen database of fingerprint images in the US Office of Personnel Management breach of 2015. This chapter proposes a distributed identity authentication protocol, called the Horcrux protocol, in which there is no such single point of compromise. The protocol relies on two standard efforts, the IEEE 2410-2017 Biometric Open Protocol Standard (BOPS) and the decentralized identifiers (DIDs) standard which is under development by the W3C Verifiable Claims Community Group. To accomplish this, we propose specification and implementation of a decentralized biometric credential storage option utilizing the concept of self-sovereign identity using blockchains.

17.1 Introduction

The proliferation of powerful mobile computing devices such as smartphones has changed the way users access many services (such as banks, e-commerce, government service). Hence, mobile biometric authentication has been introduced to evolve identity authentication beyond the current fault password model [2, 21] to strengthen the security of transactions, reduce fraud and associated costs, and improve the user experience by eliminating the need to manage multiple passwords. The use of mobile biometrics has rapidly expanded and become more mainstream thanks to companies like Apple and Samsung, and the development of embedded fingerprint sensors for smartphones such as Touch ID. Organizations across a variety of vertical industries have deployed biometric technology to provide convenient methods for users to securely access a wide range of digital services.

A. Othman (✉) · J. Callahan
Veridium US, Boston, MA, USA
e-mail: aothman@veridiumid.com

e-mail: jcallahan@veridiumid.com

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2_17

While mobile biometric authentication has the potential to offer significant value to enterprises, most security and privacy preservation schemes are still primarily based on archaic, static models that don't work any more and it is getting worse. The latest evidence of this is recent breaches disclosed by Yahoo, Equifax, and Target stores [4] that have exposed identity information for millions of individuals. Hacking attacks are not just targeting enterprises but also Federal agencies. In 2015, hackers stole the database of fingerprint images from the US Office of Personnel Management [38]. Like other stolen identity information, the unauthorized access of biometric data (i.e., biometric images and/or feature sets) can be quite damaging to individuals due to its uniqueness and intrinsic to them. Another major privacy concern is function creep [23], where authorized agencies use biometric data for purposes beyond its original intent. For example, an agency could glean additional information, such as an individual's gender, age, ancestry origin, or link biometric databases belonging to different applications. De-identifying biometric data prior to storage is an approach that has been proposed in the literature [23, 30, 31] to ensure that the stored biometric data is used only for its intended purpose and to prevent an adversary or an administrator from viewing the original identifiable data. De-identifying [27] involves storing a transformed or modified version of the biometric data in such a way that it is impossible to deduce the original biometric signal from the stored version, either an image or a feature set. However, applying a noninvertible de-identifying function on biometric images implies a loss in accuracy as discussed in [27, 30] because the transformed images are difficult to align and the discriminability of the biometric content is reduced. Meanwhile, the security of de-identified feature sets approaches [31] relies on the assumption that the key and/or the transformation parameters are known only to the legitimate user. Maintaining the secrecy of those keys is one of the main challenges since these approaches are vulnerable to linkage attacks where the key or the set of transformation parameters along with the stored template are compromised.

As such, additional research efforts must be made to keep this data secure and confidential by improving these de-identifying approaches for biometric images and feature sets. Until then enterprises will continue to store the biometric data in either the mobile or server without de-identifying it. These data storages are done within encryption layers including choosing a proper compliance system and infrastructure, which considers the particularly sensitive nature of biometric data. Nevertheless, with the news of stolen and hacked biometric data from phones [42], as well as servers breaches [38], means these schemes of storage are not the best solution.

A possible solution is the distributed storage model that has been included in the IEEE Biometrics Open Protocol Standard 2410-2017, or BOPS [1]. BOPS supports a storage model, which is neither device- nor server-centric storage [36], where the user's biometric template is distributed using a secret sharing scheme between the user's mobile device and the service provider. Both shares of the biometric data are encrypted, and for the authentication process to be successful, both shares are required [1].

However, nowadays, the user who used to consume locally installed applications on a single device or phone has moved well beyond that to an ecosystem where users own several devices and are externally authenticating their identities significantly more often. This new ecosystem requires biometric authentication models to evolve to link users to verified identity claims on the growing number of personal connected devices. This has led to the development of a series of identity management models, such as *self-sovereign identity*.

Self-sovereign identity (SSI) is a new decentralized ecosystem for private and secure identity management that is being implemented by several projects [3, 17, 32] as the replacement for traditional identity proving systems. Self-sovereign identity puts end-users—not the organizations that traditionally centralize identity—in charge of decisions about their own privacy and disclosure of their personal information and credentials. Self-sovereign identity utilizes distributed ledgers (DLT), i.e., blockchain technology, to establish a web-of-trust [6].

In this chapter, we discuss the specification and implementation of our Horcrux protocol that combines the decentralized self-sovereign identity ecosystem with 2410-2017 IEEE Biometric Open Protocol Standard (BOPS) [1].[1] Horcrux protocol is a secure and robust identity authentication solution capable of supporting different business requirements as well as the privacy of users by allowing them to manage the storage and access of their personally identifying information (PII)[2] via a distributed mobile biometric authentication system. This marriage of these two models (SSI and BOPS) via the Horcrux protocol will guarantee the following principles:

- *Existence*: Users must have an independent existence that can not only exist wholly in the digital form, and by using a biometric-based protocol , i.e., BOPS [1], for enrolling and authentication, this guarantees that the digital identity has been created and will always be verified by an existing end-user.
- *Control*: Users must control the storage and access to their identities. Under the self-sovereign identity ecosystem, users are always able to refer to, update, or even hide their personal information and credentials. The Horcrux protocol will assure that the access is always secure by their biometric which also is securely stored via the decentralized ecosystem, along with their personal information.
- *Portability and interoperability*: BOPS [1] and self-sovereign identity [32] have been designed around these principles.
- *Protection*: The security of the Horcrux protocol is trusted because it is based on strong cryptography and governed by self-sovereign identity using a blockchain technology and BOPS.

The rest of the chapter is organized as follows. In Sect. 17.2, we discuss the biometric authentication standard BOPS and the unique way to store biometric data in a distributed matter to preserve the privacy and security of the stored biometric data.

[1] The term "Horcrux" comes from the Harry Potter book series in which the antagonist (Lord Voldemort) places copies of his soul into physical objects. Each object is scattered and/or hidden to disparate places around the world. He cannot be killed until all Horcruxes are found and destroyed.

[2] Personally identifying information are data about an individual which considered to be sensitive and thus subject to security and privacy protections such as biometric and demographic data.

Section 17.3 gives a quick overview of the different identity models and evolution of these models into the new self-sovereign identity ecosystem that provides users with full control over their identity access and storage. Section 17.4 looks at the Horcrux protocol where both BOPS and SSI model can be deployed together to provide a new way for users to establish a portable, secure, and controllable biometric-based identity system which is intrinsically theirs. Finally, Sect. 17.5 summarizes the chapter.

17.2 IEEE Biometric Open Protocol Standard (BOPS) Storage Model

In traditional authentication systems such as password and PIN, only one centralized database stores the data used for authentication. When the user offers the requested proof of identity, the authentication server evaluates this proof and grants access to the user. While most security experts and enterprises see the benefits of biometric-based mobile authentication in comparison to knowledge-based systems (usually, password and PIN), the underlying architecture with which to implement biometrics is still the same centralized storage model,[3] more specifically, whether a server or mobile-centric storage approach. The following describes the server- and mobile-centric approaches. Then we describe the distributed storage model that has been adopted by IEEE BOPS.

17.2.1 Server-Centric Approach

In this setup, biometric identity data is captured by trusted means and then stored centrally on a secure server. The server-centric biometric authentication architecture is managed by the service provider. To perform a user verification, the captured biometric sample is sent to the server for processing and matching against the enrolled data stored centrally.

A server-centric approach is likely preferred for organizations that desire a high degree of control over the end-to-end process of biometric authentication and to manage and secure the storage and use of the biometric data.

This approach also supports users accessing digital services via a wide range of endpoints such as computers, mobile devices, smart TVs, and physical locations (bank branch, enterprise access control, and in-store retail scenarios). Organizations can also analyze the biometric data they collect to improve the performance of matching algorithms.

Finally, by storing more resources and functions in the cloud rather than on the device itself, it reduces app size and complexity. As a result, server-centric authen-

[3] The enrollment stage of most of the deployed biometric systems generates a digital representation of an individual's biometric trait that is stored in the system storage database [15].

tication may also function more effectively with devices that have limited memory and processing power.

Comments on Server-Centric Approach

The major concerns with this approach are security and privacy. A server-based biometric database becomes a "honeypot" target for criminals, hostile governments, and hacking groups. As the 2015 OPM hack [38], which led to the theft of millions of US government personnel fingerprint data, demonstrated storing peoples' biometric data in network accessible databases can lead to wide-scale theft of sensitive data. Furthermore, there is the privacy concern of function creep where the biometric data is used in different purposes than authentication such as improving matching algorithms, databases linkage without consent, and deriving additional demographic information [23, 27].

Moreover, it is a generally accepted privacy principle that individuals must be able to access their PII and update it where necessary; therefore, some jurisdictions have already specifically referenced biometric data in privacy guidance and legislation such as European General Data Protection Regulation (GDPR) [10].

GDPR is European Union's new set of policies on data protection that officially took effect on May 2018. While this regulation focuses on the citizens of European Union (EU), and reshapes the way organization across Europe handle citizens' PII data, any organization outside of EU that collects or processes data of EU citizens is also affected. GDPR expressly identifies biometric data as a category of sensitive personal data and requires the development of solutions with adequate privacy measures in place giving individuals' choice and control of their data. This means that organizations must ensure that individuals can access their biometric data as and when they request it. Further, organizations must have processes in place to allow individuals to correct, update, and delete their data where necessary.

Based on such data privacy regulation, compared to server-centric storage of biometric data, the storage and matching of biometric data on smartphones for authentication purposes are compelling and more straightforward approaches to satisfy global privacy requirements.

17.2.2 Mobile-Centric Approach

In this setup, biometric template creation, storage, and matching all occur locally on the device which allows an organization such as a bank to enable strong biometric authentication into their mobile app without having to manage PII on a central server. The mobile-centric biometric systems are getting growing support for solutions which are incorporating FIDO authentication protocols [11]. In a FIDO-compliant system, a successful biometric match grants access to a private key stored on the device, which is in turn used to respond to a public key infrastructure (PKI) challenge[4] [28] from a relying party, such as a bank or retailer whose app is running on the device.

A mobile-centric approach is likely the best option for organizations with a primary objective of preventing large-scale breaches of customer data and satisfying global privacy requirements. Storing and matching biometric data on a device gives users more control over their data.

The mobile-centric approach for storing biometric data is also gaining momentum because now most major smartphone manufacturers are shipping devices that support biometric authentication and providing access to third parties via APIs. These advances are enabling organizations to swiftly roll out mobile-based biometric authentication services. Therefore, this mobile-centric model is being adopted by organizations, including banks and payment service providers (PSPs), as a quick way of solving the "password" problem.

Comments on Mobile-Centric Approach

The manufacturer-led, mobile-centric model only solves part of the problem of providing secure and convenient access. Organizations are still looking at alternatives to ensure that an authentication solution is available to a large percentage of their users' base. The mobile-centric approach only offers biometric authentication to those equipped with the latest mobile devices with integrated biometric sensors and secure hardware to store sensitive biometric data. In addition, mobile biometric apps are consuming more disk and runtime footprints since the biometric processes all take place on the app, which less powerful devices may not easily support.

Moreover, as the data remains on the device, there are no transfers of the biometric data unless users perform backups to the cloud to avoid re-enrolling in cases of lost or damaged devices. However, most of the organizations that adopt the mobile-centric approach do not provide such backup services.

Finally, there are genuine concerns for organizations operating in highly regulated sectors, such as finance and health care, that this model to capture and store biometric data is managed by smartphone manufacturers using algorithms tuned to be more convenient than secure.

Although most of these deployed mobile biometric authentication systems by manufacturers are applying mechanisms to protect the integrity and confidentiality of data storage and code execution (i.e., Trusted Execution Environments [8] and Secure Elements [34]), Zhang et al. [42] revealed some severe issues with one of the deployed Android fingerprint frameworks which is using an embedded fingerprint sensor. They exploited an HTC One device with malware and demonstrated that an attacker can collect fingerprint images of victims every time they swipe their fingers.

[4] The private key is used to respond to the PKI challenge and never leaves the mobile device.

17.2.3 BOPS Distributed Storage Approach

The choice of either a device- or server-centric biometric authentication method provides organizations with both positive and negative consequences. However, the main concern with both approaches is that there is a single point to compromise biometric data.

There is, however, a third approach that is a privacy-centric and also provides service providers with a mechanism of managing the storage of their customers/employees data without relying only on the operating system provided by a device manufacturer. This model is a distributed storage model that has been introduced by Othman and Ross [27] and adopted by the Biometrics Open Protocol Standard, or BOPS, which is IEEE standard 2410-2017.

The IEEE 2410-2017 Biometrics Open Protocol Standard (BOPS) [1] demands high levels of assurance to control communication between an organization server and its clients via two-way secure socket layer/transport layer security (SSL/TLS) and to monitor authentication logs and patterns with enhanced intrusion detection system (IDS) analytics.

The difference between BOPS approach and the aforementioned approaches (server- or mobile-centric) that the biometric enrolled data, i.e., representation of a fingerprint, voice, facial features, is cryptographically protected into two shares using a secret sharing scheme, i.e., visual cryptography [25]. These encrypted shares are stored, respectively, on a client device and a remote BOPS server, such that the biometric data is not kept in a single point to compromise.

Visual cryptography scheme [25] (VCS) is a simple and secure way to share a secret such that decryption can be performed using a simple binary operation. The basic scheme is referred to as the k-out-of-n visual cryptography scheme which is denoted as (k, n) VCS [25]. Given an original binary data T, it is encrypted into shares such that:

$$T = S_{h_1} \oplus S_{h_2} \oplus S_{h_3} \oplus \ldots \oplus S_{h_k} \tag{17.1}$$

where $\oplus$ is a boolean operation, $S_{h_i}, h_i \in 1, 2, \ldots, k$ is a share which appears as white noise image, $k \leq n$, and n is the number of these shares. It is difficult to decipher the secret T using individual S_{h_i}'s [25]. The encryption is undertaken in such a way that k or more out of the n generated shares are necessary for reconstructing the original secret T.

As shown in Fig. 17.1, BOPS defines three steps during enrollment. First, the remote server generates a public–private key pair (RKP) in which the public key is sent to the mobile device. Then, a biometric template (called the initial biometric vector or "IBV") is collected, encrypted into two shares (shares I and II) using 2-out-of-2 scheme, and then paired with a device-generated public–private key pair (LKP). In the third step, the LKP private key is reserved locally and the LKP public key along with the biometric share II is encrypted with the RKP public key for transmission to the server over a two-way TLS connection and IBV is discarded. The client cer-

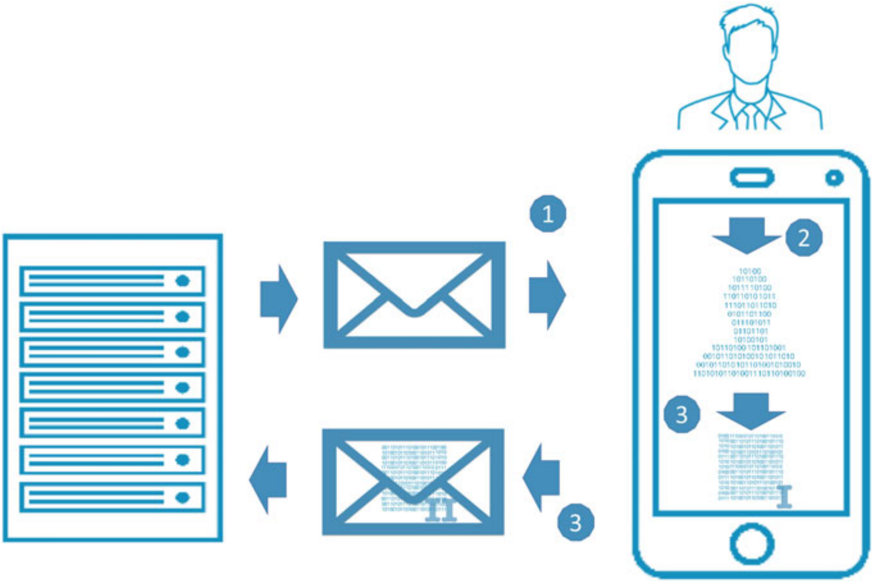


Fig. 17.1 Illustration of distributed model steps during the biometric enrollment stage. (1) Server sends an enrollment request along with the RKP public key, (2) biometric capture is encrypted into two shares (I and II) using visual cryptography, and (3) biometric share II and device (LKP) public key are encrypted by the server (RKP) public key and sent to the server via two-way TLS

tificate for the TLS connection is installed a priori via application installation on the mobile device.

During authentication, a candidate biometric vector (CBV) is acquired for matching with IBV. BOPS defines two configuration modes for authentication:

- *Local Match*: This configuration is used in the case the biometric matching is done on mobile devices. The server is requested to encrypt (using its RKP private key) IBV share II it holds and returns them to the local device. The CBV is collected, IBV shares from local (I) and remote (II) combined and matched on the local device. The CBV and combined IBV are subsequently wiped from volatile memory.
- *Remote Match*: This configuration is used in the case the biometric matching is done remotely on the server. The collected CBV and the local IBV share I are encrypted in an envelope with the RKP public key and transmitted to the server. On the server, the incoming IBV share from the local device is combined with server-based share and compared to the incoming CBV. The CBV and combined IBV are subsequently discarded.

The choice between these two different configurations is set according to the policy of the enterprise that deploys BOPS in their biometric authentication framework.

A distributed storage approach combines convenience, personal privacy, and enhanced security to create a model that makes it harder for attackers to compromise a system.

The fundamental idea of this distributed approach is utilizing secret sharing scheme [25] that, rather than encrypting the data as a single file using the standard public and private key pairing methodology, biometric data is encrypted randomly into multiple shares. These shares must be combined in order to recreate the original biometric data, ensuring that only the people, or devices, that possess the encrypted share files are able to recombine them and gain access to the protected information without any influence to the overall matching performance. Therefore, in the BOPS model, if the central biometric database, i.e., server is hacked, then attackers still need to have the user device's share of the biometric vector to break the system. Conversely, if a user has their mobile device compromised, an attacker still needs to break into the central database. This ensures that the biometric data is protected from data breaches, provides peace of mind for the end-user that their biometric cannot be easily compromised, and enhances the storage architecture to eliminate misuse of the data. Moreover, this distributed model has two different matching configurations which allow an organization to customize their solutions based on their customer-base used technologies and network connectivity.

Hence, this simple IEEE open protocol standard solves the single point of failure and control concerns with a storage model that can lead to the deployment of more secure, flexible, and interoperable biometric authentication solutions. Table 17.1 provides a high-level summery of discussed approaches for biometric data storage.

Comments on BOPS Distributed Model

Today, technology vendors and organizations have employed vastly different approaches when building biometric technology solutions to address identity and access problems. It means that user PII exists within the context of each specific Web site or application they use and control over their identity and data must be exerted on a site-by-site, app-by-app basis. So, even if some organization are adopting the IEEE BOPS model for storage, users still have the burden to manage and consolidate their digital identity that is scattered across different organizations, with no ability to secure them effectively if organizations are adopting different approaches to manage biometric data (i.e., server- or mobile-centric model).

At the same time, there is a growing inefficiency when organizations all around the world have to collect, store, and protect the same sort of PII (either biometric, demographic data or personal credentials) in their own silos and most of these silos are reaching their tipping points.

Hence, fundamental changes are needed to evolve identity authentication in order to improve the user experience by eliminating the need to manage multiple accounts and to eliminate reliance on any one vendor or group of vendors.

We believe the common denominator across most aspects of personally identifying information (PII) protection is identity. An identity is inextricably linked to a person, device, application, system, or network, and it is the most dependable 'perimeter' we can rely upon to determine how to make information available securely and adequately.

Table 17.1 A high-level comparison of different models to storage biometric data

Biometric storage	Pros	Cons
Server-centric model	• High degree of control over the biometric authentication to manage and secure the data storage • Multiple end-point access and services using the same biometric authentication system • Analyze the biometric to improve the performance of matching algorithms • Suitable for identification scenarios (1: N matching) • Reduce app size and complexity of terminal access point of the service	• Large-scale data breaches • Privacy concerns of misusing and cross-linking the biometric data without users' consents
Mobile-centric model	• No managing and storing of PII or biometric data in a central database • No privacy concerns of misusing the data • Taking advantages of recent devices that have shipped with biometric capabilities	• Biometric authentication only available to owners of the latest mobile devices with biometric capabilities • Lost and stolen devices are problematic scenarios • Most of the available mobile devices are built to be convenient more than secure
BOPS distributed model	• No single point to compromise • A privacy-centric approach where the organizations don't have to manage or to store biometric data • Taking advantage of a secret sharing scheme to securely store the biometric data in a distributed manner • Biometric trait-agonistic approach • Supported by IEEE 2410-2017 standard that guarantees high levels of assurance to control communication between an organization server and its clients and to monitor authentication logs and patterns with an enhanced intrusion detection system • Matching can be configured to be on mobile or server based on enterprises' policies	• Cannot be utilized for identification scenarios (1: N matching) • The users have the burden to manage and consolidate their digital identity that is scattered across different organizations

In the following sections, we give a quick overview of the different identity models and evolution of theses models into the new self-sovereign identity ecosystem that provides users a full control over their identity accessing and storage. Then, we discuss our Horcrux protocol where both BOPS and SSI model can be deployed together to provide a new way for users to establish a portable, secure, and controllable biometric-based identity system which is intrinsically theirs.

17.3 Self-sovereign Identity Ecosystem

Current identity proving methods (see Fig. 17.2) rely on specific parties: an *issuer*, *end-user*, *verifier*, and *inspector*.

Issuers such as governments associate identity credentials with end-users. Then, the issuer shares personal information and credentials of the end-user with a verifier. If the end-user applies for a bank account, credit card, or car loan, the inspector contacts a verifier to prove the claimed identity by the end-user. Therefore, especially if this process is online, the inspector presents a multiple-choice quiz about past addresses or who financed the user's last car. That's an identity verification service that a verifier provides to lenders and others, i.e., inspectors. Based on the answers or proof of holding the credentials, the inspector will verify the claimed identity by the end-user and guarantee the required service. This ecosystem has the same security flaw as the traditional authentication systems; end-user personal data (e.g., SSN, addresses, birthdate) is stored in a centralized database of the verifier. An example of this security flaw is the data breach of a verifier in USA; Equifax [14].

In current digital and interconnected practice, these verifiers become a centralized database which stores the data used for authentication. When the user offers the requested proof of identity, the authentication server evaluates this proof and grants access to the user. For example, when a user tries to access his account on a typical Web application, he is prompted to enter a password. Traditionally, the Web application holds the information about the user's account and his password. When the user submits his password during log-in process, the application compares the stored password to the submitted password. If they match, the user is granted access to the application. In other words, all the information needed to authenticate the user is held on a single system. This silo-based approach, where users must maintain identities for every site they interact with, has become untenable. It is not just a usability disaster for individuals, it also creates a multitude of data honeypots for hackers which when breached, compromises trust in all Internet services.

To solve this problem, in some current implementations, the authentication server can be completely separated from the server running Web applications or biometric authentication database . For example, single sign-on (SSO) schemes [29] are based on this concept. SSO schemes rely on a third-party identity provider (IdP) to broker authentication using protocols such as SAML [13] and OpenID Connect [37]. Since their introduction in 2002 and 2010, respectively, only 5% of sites use any of over

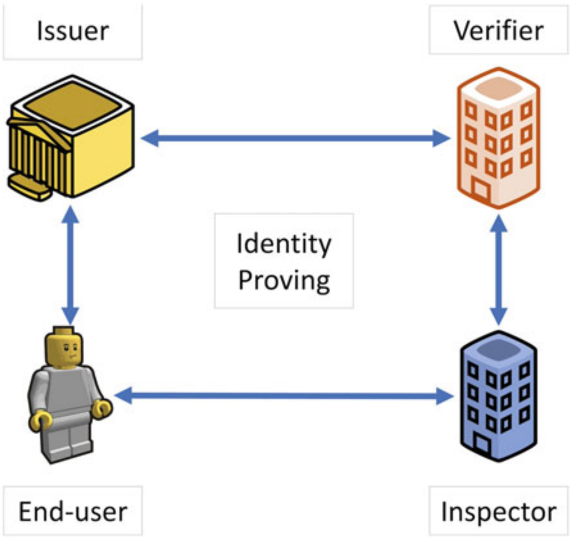


Fig. 17.2 Traditional identity proving ecosystem

50 disparate IdP [41] SSO services (e.g., “Login with Facebook” and “Sign in with Google”).

However, these have produced inadvertent side effects such as concentrating control around a small number of providers, increasing data leakage through inadvertent sharing, and raising privacy concerns, all while not actually giving the individual real control.

Surveys of users show an overwhelming dissatisfaction with single sign-on (SSO), a feeling of “lack of control” over their data [20, 35, 39] and a desire to control it themselves. Recent legislation, such as the General Data Protection Regulations (GDPR) [10, 16] and Payment Services Directive II (PSD2) [7], are pressuring institutions, both private and public, to place citizen or customer data into the end-user’s control.

Self-sovereign identity is a new identity ecosystem where individuals (or even organization) control, and manage their identities. In this sense, the individual is their own identity provider—no external party can claim to “provide” the identity for them because it is intrinsically theirs. In other words, self-sovereign identity is as a digital record or container of identity transactions that end-users control. The end-user can add more data to it, or ask others to do so, reveal some the data or all of it some of the time or all the time. Moreover, end-users can record their consent to share data with others and easily facilitate that sharing. It is persistent and not reliant on any single third party. Claims made about an end-user in identity transactions can be self-asserted or asserted by a third party whose authenticity can be independently verified by a relying party. The infrastructure of self-sovereign identity has to reside in an environment of diffuse trust which is not controlled by any single organization or even a small group of organizations. The cryptographically secure blockchain is the breakthrough technology that makes this possible. It enables multiple entities such as organizations and governments to cooperate mutually via distributed consensus to form decentralized blockchains, where data is replicated in multiple locations to be resistant to faults and tampering. While distributed ledger technology has been around for some time, new blockchain applications, such as Bitcoin, have resulted in realizations of its potential, particularly with respect to decentralization and security.

17.3.1 Distributed Ledgers Technology (DLT)

A DLT is a cryptographically secure, decentralized, and distributed ledger of information. In this chapter, we use the term DLT and blockchain interchangeably to encompass all implementations of such architecture. DLT replace trust in humans with trust in mathematics via cryptographic triple play [18]:

(1) Each transaction in the blockchain is digitally signed by the originator.
(2) Each transaction—singly or in blocks—is chained to the prior via a digital hash.
(3) Validated transactions are replicated across all machines using a consensus algorithm.

The result is an immutable time-stamped append-only distributed ledger, which contains a set of cryptographically hashed transactions. When a new record/transaction is added to a chain, all other distributed instances of the chain are updated with the new record. This provides complete transparency of every transaction that makes it very difficult, if not almost impossible to change past transactions or maliciously control future ones.

DLT implementations can be divided into two categories [18]: public and private. A public DLT typically has a lower transaction throughput as the network is public and larger; therefore, the consensus mechanism requires more time and resources. Conversely, in a private DLT, only a selection of verified entities has the privilege to access the ledger, and consequently, the transaction approval rate is higher. A private DLT provides more privacy but less transparency on the content of the transactions. These public or private DLTs can be either permissionless and permissioned. In a permissionless DLT, any entity can theoretically participate in writing into the ledger. In a permissioned blockchain, however, only the authorized entities are permitted to participate in validating and adding transactions to the ledger.

These DLT different implementations underpin the majority of cryptocurrencies by allowing for transactions to take place without central intermediaries [22, 40]. Bitcoin [24] and Ethereum [9] are common examples of blockchain-based cryptocurrencies. However, DLT use cases have expanded beyond the financial services industry.

DLT is leading the next evolution of the identity by the creation of a common identity layer that allows people, organizations, and things to have their own self-sovereign identity (SSI)—a digital identity they own and control, and which cannot be taken away from them. The decentralized nature of the blockchains networks ensures data integrity and availability, as well as privacy for the users, as there is no need for the continuous involvement of the identity issuer, for identity access, resolution, or verification.

17.3.2 Self-sovereign Identity Ecosystem Architecture

Figure 17.3 provides an overview of the self-sovereign identity architecture. The followings are the brief descriptions of the architecture entities. Note that in this architecture, the information is no longer centralized and connections are individually permissioned.

- *DID:* Decentralized Identifiers (DIDs) are a new type of identifier intended for a self-sovereign identity system, i.e., entirely under the control of an entity and not dependent on a centralized registry or certificate authority. DIDs are opaque, unique sequences of bits, that get generated when a user accepts a claim from an issuer along with a corresponding DID document. DIDs have a foundation in Universal Resource Identifiers (URIs) [19, 33]; therefore, they achieve global uniqueness without the need for a central registration authority.

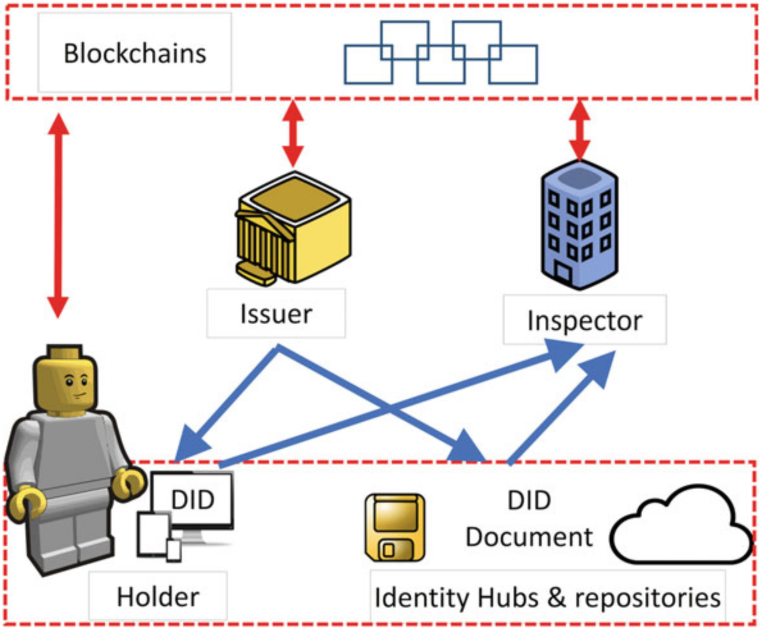


Fig. 17.3 Self-sovereign identity ecosystem architecture

- *DID document:* A DID resolves to a corresponding DID document—a simple document that contains all the metadata needed to interact with the DID. Specifically, a DID document typically contains at least three things along with personal information or credentials. The first is a set of mechanisms that may be used to authenticate as a particular DID (e.g., public keys, biometric templates, or even an encrypted share of biometric data). The second is a set of authorization information that outlines which entities may modify the DID document. The third is a set of service endpoints, which may be used to initiate trusted interactions with an entity [33].
- *Blockchains:* In this architectural construct, the blockchain acts as an index of identifiers and audit trail of permissioned exchanges between the issuer of claims, the holder of claims, and the inspector of claims.
- *Identity hubs and repositories:* These hubs are secure personal data repositories that curate and coordinate the storage of signed/encrypted DID documents, and relay messages to identity-linked devices. Examples of identity hubs include Dropbox, Google Drive, and Storj.
- *Issuer:* An entity that creates DID and DID documents associates it with a particular subject and transmits it to a holder. Examples of issuers include corporations, governments, and individuals.
- *Inspector/Verifier:* Inspectors request claims in the form of DIDs from subjects and organizations in order to give them access to protected resources. The inspector verifies that the credentials provided via DID and in the DID document are fit-for-purpose and also checks the validity of the DID in the blockchain. Examples of inspectors include employers, security personnel, and Web sites.
- *Holder:* Holders receive DIDs from issuers, store DID documents via identity hubs, and provide DID documents to inspectors. The entity which controls a particular DID can be the subject of the DID document, but not necessarily. An inspector can also resolve DIDs into their corresponding DID documents and discover DIDs across a decentralized system. Examples of holders are users—students, employees, and customers. Other examples of holders that have the permissions to handle subject's claims include Web services or mobile apps installed on the subject's personal devices.

SSI users have the liberty to manage their identity data on their mobile devices or cloud repositories. Mobile devices have become an essential part of our lives. We use mobile devices to store our credentials and payment. Therefore, while physical documents and storage of identity attributes on the cloud and third-party identity providers may exist for the years to come, storing identity data on mobile devices is the next natural step toward the realization of self-sovereign identity and using mobile biometric can help in facilitating this to protect and authenticate digital identities.

17.4 The Horcrux Protocol

The IEEE 2410-2017 standard, BOPS allows for interoperability at several layers including the persistence cluster [1] provided it satisfies security requirements for storage of encrypted biometric shares. We propose any BOPS server can act as a *holder* of biometric shares via blockchain using methods outlined in the W3C Decentralized Identity (DID) specification [33]. A BOPS server can enroll a user by storing biometric share(s) as DID documents using off-chain storage providers owned by the user. The corresponding DID acts as the identity assertion associated with the enrolled biometric.

17.4.1 Enrollment

Figure 17.4 depicts a standard BOPS enrollment flow (adapted from [1] Sect. 7.2). The user (via a browser user-agent) is prompted to enroll their biometrics with a service provider acting as an *issuer*. The initial biometric vector (IBV) is encrypted (via visual cryptography) into two shares. One share is reserved on the local mobile device while the second is transmitted to the BOPS server. Instead of a persistence cluster (e.g., SOLR) backend, the BOPS server relies on a blockchain store in this case using a decentralized identifier (DID) [33] for persistence. DIDs provide a blockchain-agnostic method for resolving DID documents much like URIs [19] uniquely characterize Web resources via URNs and URLs, but for disparate blockchain ecosystems. The W3C Verifiable Claims Community Working Group has defined DID method specifications [33] for implementors of CRUD[5] operations specific to a particular blockchain. The BOPS server acts as a resolver given a DID to fetch the corresponding DID document if possible. The DID and corresponding DID document are cryptographically associated with each other via blockchain transactions such that any tampering with the DID document for a given DID would be evident. After persisting the DID document and registering the associated DID on a blockchain, the user is notified of success (or failure) of their enrollment. It should be noted that no biometric shares are stored on any blockchains, only in DID Documents that are persisted "off-chain" via identity hubs or personal storage providers.

17.4.2 Authentication

The encrypted biometric share is still within an encrypted envelope as per [1], but the share is persisted on a corresponding DID document of an associated DID. The DID can be used as a claim with another BOPS server acting as a *verifier*. Again, this is

[5] In computer programming, create, read, update, and delete (CRUD) are the four basic functions of persistent storage.

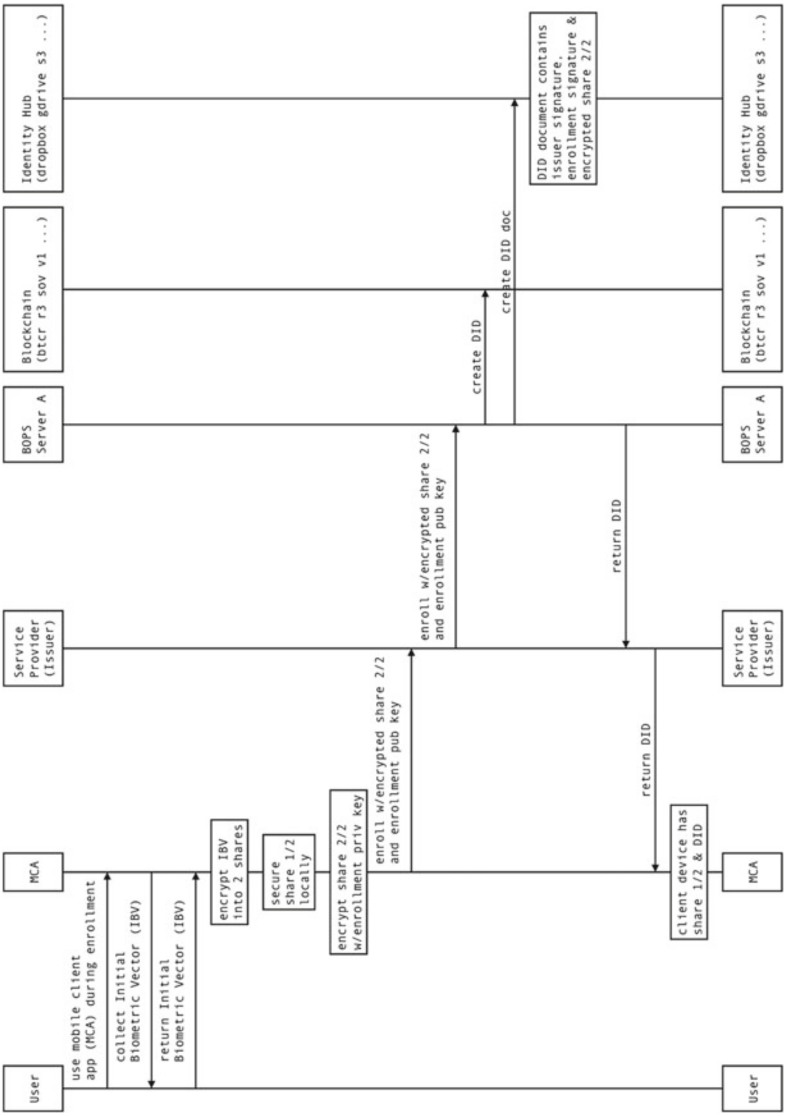

User
MCA
Service Provider (Issuer)
BOPS Server A
Blockchain (btcr r3 sov v1 ...)
Identity Hub (dropbox gdrive s3 ...)
use mobile client app (MCA) during enrollment
collect Initial Biometric Vector (IBV)
return Initial Biometric Vector (IBV)
encrypt IBV into 2 shares
secure share 1/2 locally
encrypt share 2/2 w/enrollment priv key
enroll w/encrypted share 2/2 and enrollment pub key
enroll w/encrypted share 2/2 and enrollment pub key
create DID
create DID doc
DID document contains issuer signature, enrollment signature & encrypted share 2/2
return DID
return DID
client device has share 1/2 & DID
User
MCA
Service Provider (Issuer)
BOPS Server A
Blockchain (btcr r3 sov v1 ...)
Identity Hub (dropbox gdrive s3 ...)

Fig. 17.4 Enrollment sequence

possible because any tampering with the DID document associated with a given DID will be detectable due to their relationship via a recorded blockchain transaction [33]. Figures 17.5 and 17.6 show examples of a different BOPS server being used by a verifier (BOPS server B), where the user tries to access a resource on a Web site (e.g., the service provider) using a mobile client application (MCA) with a DID created by an issuer 17.4 and a public key created at enrollment. The service provider relies on a BOPS server to resolve the DID and fetch the corresponding DID document via a blockchain from the storage provider. If the DID document is a valid claim, the BOPS server checks if the issuer of the claim is known (via its public key in the DID document) and that the enrollment public key matches for this user as well. If valid, the user (via their MCA) is requested for their candidate biometric vector (CBV), i.e., probe and complement share of the IBV as per [1].

17.4.2.1 Remote Authentication

In the case of remote authentication/match, upon receiving the complementary share and CBV from the client (as described in 17.2—Remote configuration mode), the enrollment public key is used to decrypt the client's share, combine the IBV shares, and match them to the CBV. If successful, the user is authenticated. Note that, the service provider, acting as a verifier, uses a different BOPS server instance to authenticate the user even though this user has never registered at this service provider. Furthermore, the user and service provider are the only parties needed at authentication time unlike SAML or OAuth that rely on third-party identity providers (IdPs) to broker identity claims in traditional single sign-on (SSO) systems. The Horcrux protocol supports *self-sovereign identity* [5] by using blockchain technology to secure credentials issued by valid authorities (i.e., *issuers*) for later use directly by the user who owns the credentials. The user may store such credentials via several personal cloud storage providers such as Dropbox, Google Drive, and Amazon S3. But the user delegates management (via OAuth tokens) to a *holder* such as the BOPS server. The holder can access issued claims like the encrypted biometric shares on behalf of the user during authentication, but require biometric authentication as specified in the `authenticationCredentials` section of the claim [33].

17.4.2.2 Local Authentication

The local configuration matching mode of BOPS is also available such that a combination of biometric shares occurs on the mobile device. Figure 17.6 shows this variation in which the second biometric share is retrieved via DID referencing from the corresponding DID document but is transmitted to the client by a service provider and its BOPS server. The biometric share is opaque to the service provider and BOPS server in this case, but the server knows that the corresponding share on the mobile device is used for matching due to the use of Hash-based Message Authentication

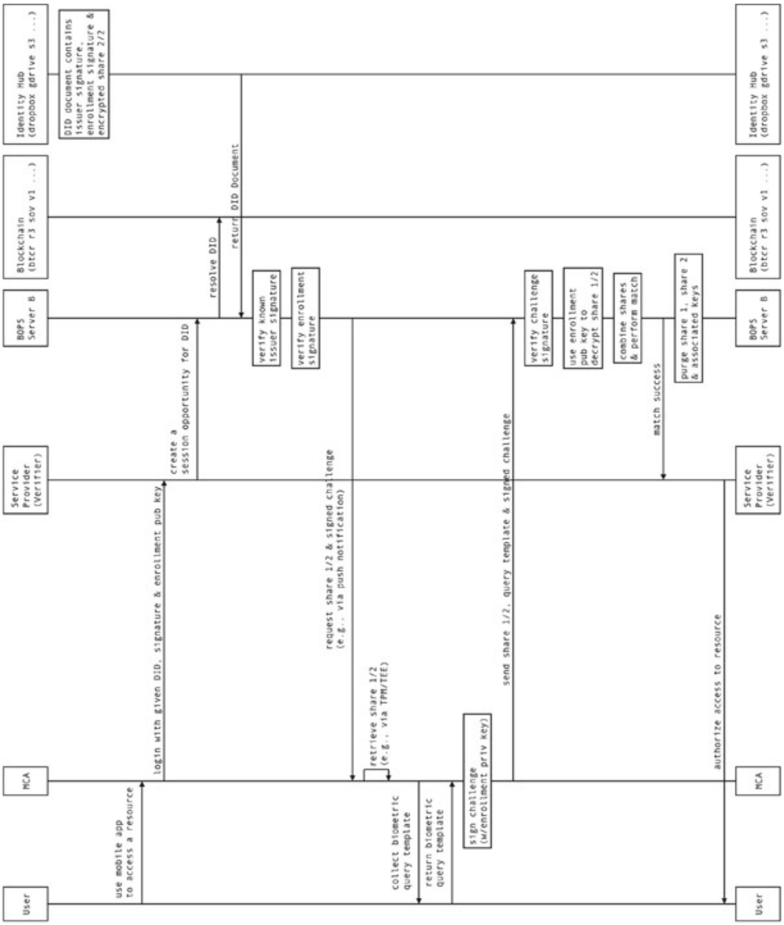

User
MCA
Service Provider (Verifier)
BOPS Server B
Blockchain (btcr r3 sov v1 ...)
Identity Hub (dropbox gdrive s3 ...)
DID document contains issuer signature, enrollment signature & encrypted share 2/2
use mobile app to access a resource
login with given DID, signature & enrollment pub key
create a session opportunity for DID
resolve DID
return DID Document
verify known issuer signature
verify enrollment signature
request share 1/2 & signed challenge (e.g., via push notification)
retrieve share 1/2 (e.g., via TPM/TEE)
collect biometric query template
return biometric query template
sign challenge (w/enrollment priv key)
send share 1/2, query template & signed challenge
verify challenge signature
use enrollment pub key to decrypt share 1/2
combine shares & perform match
match success
purge share 1, share 2 & associated keys
authorize access to resource
User
MCA
Service Provider (Verifier)
BOPS Server B
Blockchain (btcr r3 sov v1 ...)
Identity Hub (dropbox gdrive s3 ...)

Fig. 17.5 Remote authentication sequence

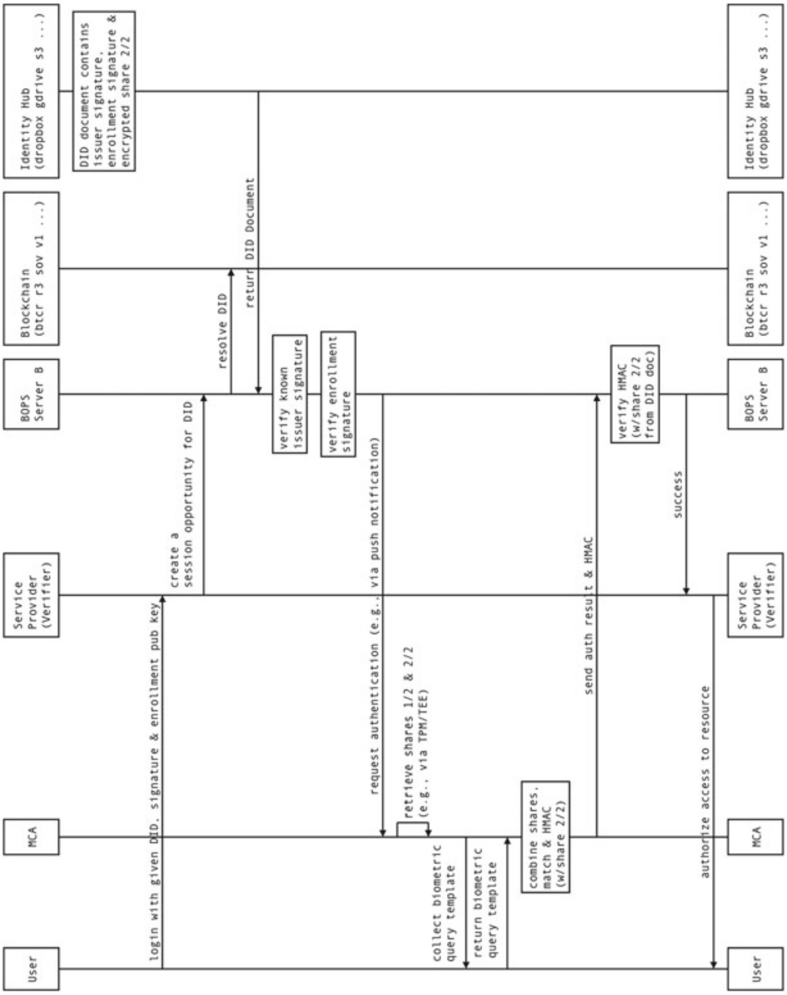

User
MCA
Service Provider (Verifier)
BOPS Server B
Blockchain (btcr r3 sov v1 ...)
Identity Hub (dropbox gdrive s3 ...)
DID document contains issuer signature, enrollment signature & encrypted share 2/2
login with given DID, signature & enrollment pub key
create a session opportunity for DID
resolve DID
return DID Document
verify known issuer signature
verify enrollment signature
request authentication (e.g., via push notification)
retrieve shares 1/2 & 2/2 (e.g., via TPM/TEE)
collect biometric query template
return biometric query template
combine shares, match & HMAC (w/share 2/2)
send auth result & HMAC
verify HMAC (w/share 2/2 from DID doc)
success
authorize access to resource

Fig. 17.6 Local authentication sequence

Code(HMAC)[6] while retrieving and sending the encrypted second share by BOPS server (BOPS B in Fig. 17.6). The enrolled share is never sent to the server, but both shares are kept locally as per BOPS local configuration mode. The matching is done locally and authentication decision. MCA sends the decision after computing an HMAC using the share and sends it to the server. The server can compare the HMAC key with the opaque encrypted share from the DID document. It is possible, however, that the user could resolved a given DID, retrieve the corresponding DID document, extract the opaque encrypted share, and construct the HMAC, thus spoofing possession of that share and falsifying the biometric match. We are in the process of investigating methods for securing DIDs on a mobile device and/or using server-based key mechanism to prevent this attack vector.

17.5 Summary

The threat of cyber attacks and the explosive growth of mobile and connected devices has ignited the quest for practical, secure, and privacy-preserving digital identity and access management (IdM) architectures with highly secure authentication solutions. While the self-sovereign identity (SSI) model is the next evolution of identity management paradigm in which users have complete ownership and control over their digital identity, there is the need to provide the users with a secure, reliable, and interpretable biometric authentication model to control the storage and access to their digital identities. The Horcrux protocol is a method for secure exchange of biometric credentials within an existing standard (IEEE 2410-2017 BOPS [1]) implemented across next-generation blockchain-based self-sovereign identity platforms based on open standards like DIDs and DID documents [33]. By using blockchain and off-chain storage as an alternative to the persistent layer in BOPS, we use new blockchain-agnostic standards to enroll via an issuer and authenticate on a verifier that is not part of a real-time trust network. Instead, they rely on user-controlled biometric credentials that are cryptographically encrypted into multiple shares across the user's device and blockchain-linked personal storage providers. The protocol is generalized for two or more biometric shares that can be stored across mobile devices and personal storage providers with redundancy for availability and safety. Future plans include a reference implementation and detailed analysis of the protocol for performance and correctness using TLA+ in a manner similar to the protocol analysis of WPA found in [26]. Further, the IEEE 2410-2017 standard allows for more than two encrypted shares. Hence, as a continuation of the proposed Horcrux protocol, algorithms such as visual cryptography [36] and Naor and Shamir secret sharing [25] can be utilized for larger number of shares. Using DIDs and associated DID documents for more biometric shares across different blockchains and replicating copies of shares could considerably protect users from compromise and increase availability.

[6] HMAC is an approach to verify a message integrity by ensuring that the data has not been altered or replaced when send back to the sender [12].

Acknowledgements The authors would like to thank Ward Rosenberry for his help in editing and proofreading the chapter.

References

1. 2410-2017 IEEE Biometric Open Protocol Standard (BOPS). https://standards.ieee.org/findstds/standard/2410-2017.html
2. Adams A, Sasse MA (1999) Users are not the enemy. Commun ACM 42(12):40–46
3. Ali M, Nelson JC, Shea R, Freedman MJ (2016) Blockstack: a global naming and storage system secured by blockchains. In: USENIX annual technical conference, pp 181–194
4. Armerding T. The 17 biggest data breaches of the 21st century. https://www.csoonline.com/article/2130877/data-breach/the-biggest-data-breaches-of-the-21st-century.html
5. Baars D (2016) Towards self-sovereign identity using blockchain technology. Master's thesis, University of Twente
6. Caronni G (2000) Walking the web of trust. In: IEEE 9th international workshops on enabling technologies: infrastructure for collaborative enterprises, pp 153–158
7. Cortet M, Rijks T, Nijland S (2016) Psd2: the digital transformation accelerator for banks. J Payments Strategy Syst 10(1):13–27
8. Ekberg J-E, Kostiainen K, Asokan N (2013) Trusted execution environments on mobile devices. In: Proceedings of the 2013 ACM SIGSAC conference on computer and communications security, ACM, pp 1497–1498
9. Ethereum project. https://www.ethereum.org/
10. European Union General Data Protection Regulation (GDPR). http://eugdpr.org/eugdpr.org.html
11. FIDO UAF Protocol Specification v1.0 Proposed Standard (2014) https://fidoalliance.org/specs/fido-uaf-v1.0-ps-20141208/fido-uaf-protocol-v1.0-ps-20141208.html
12. Hash-based Message Authentication Code. https://en.wikipedia.org/wiki/HMAC
13. Hughes J, Maler E (2005) Security assertion markup language (saml) v2. 0 technical overview. OASIS SSTC Working Draft sstc-saml-tech-overview-2.0-draft-08, pp 29–38
14. Hume M (2018) Identity theft cited as threat after equifax security breach. The Globe and Mail, Toronto A, 7
15. Jain A, Ross A, Prabhakar S (2004) An introduction to biometric recognition. IEEE Trans Circuits Syst Video Technol 14(1):4–20
16. Koops B-J, Leenes R (2014) Privacy regulation cannot be hardcoded. Intl Rev Law Comput Technol 28(2):159–171
17. Lundkvist C, Heck R, Torstensson J, Mitton Z, Sena M (2016) Uport: a platform for self-sovereign identity, Draft Version (2016-10-20). http://blockchainlab.com/pdf/uPort_whitepaper_DRAFT20161020.pdf
18. Marc P (2016) Research handbook on digital transformations, Chapter Blockchain technology: principles and applications. Edward Elgar Publishing
19. Mealling M, Denenberg R (2002) Report from the joint w3c/ietf uri planning interest group: uniform resource identifiers (uris), urls, and uniform resource names (urns): clarifications and recommendations. Technical report
20. Mertens W, Rosemann M (2015) Digital identity 3.0: the platform for people. Technical report
21. Morris R, Thompson K (1979) Password security: a case history. Commun ACM 22(11):594–597
22. Mukhopadhyay U, Skjellum A, Hambolu O, Oakley J, Yu L, Brooks R (2016) A brief survey of cryptocurrency systems. In 14th annual conference on privacy, security and trust(PST), IEEE, pp 745–752
23. Nagar A, Nandakumar K, Jain A (2010) Biometric template transformation: a security analysis. In: Proceesings of SPIE, Electronic imaging, media forensics and security XII, San Jose

24. Nakamoto S (2008) Bitcoin: a peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf
25. Naor M, Shamir A (1994) Visual cryptography. In: Workshop on the theory and application of cryptographic techniques. Springer, Heidelberg, pp 1–12
26. Narayana P, Chen R, Zhao Y, Chen Y, Fu Z, Zhou H (2006) Automatic vulnerability checking of IEEE 802.16 wimax protocols through TLA+. In: 2nd IEEE workshop on secure network protocols, pp 44–49
27. Othman A, Ross A (2015) De-identifying biometric images by decomposition and mixing. In: Ngo D, Teoh A, Hu J (eds) Biometric security. Cambridge Scholars Publishing
28. Public key infrastructure. https://en.wikipedia.org/wiki/Public_key_infrastructure
29. Radha V, Reddy DH (2012) A survey on single sign-on techniques. Proc Technol 4:134–139
30. Ratha N, Connell J, Bolle R (2001) Enhancing security and privacy in biometrics-based authentication systems. IBM Syst J 40(3):614–634
31. Rathgeb C, Uhl A (2011) A survey on biometric cryptosystems and cancelable biometrics. EURASIP J Inf Secur 1:1–25
32. Reed C, Sathyanarayan UM, Ruan S, Collins J (2018) Beyond BitCoin—legal impurities and off-chain assets. Int J Law Inform Technol 26(2):160–182
33. Reed D, Sporny M (2017) W3C decentralized identifiers (dids) 1.0. https://w3c-ccg.github.io/did-spec/
34. Reveilhac M, Pasquet M (2009) Promising secure element alternatives for NFC technology. In: IEEE first international workshop on near field communication, pp 75–80
35. Rose J, Rehse O, Röber B (2012) The value of our digital identity. Boston Cons, Gr
36. Ross A, Othman A (2011) Visual cryptography for biometric privacy. IEEE Trans Inf Forensics Secur 6(1):70–81
37. Sakimura N, Bradley J, Jones M, Medeiros B, Jay E (2011) Openid connect standard 1.0. (online). http://openid.net/specs/openid-connect-standard-1_0-21.html. Accessed 30 Mar 2013
38. Sanger DE (2015) Hackers took fingerprints of 5.6 million U.S. workers, government says. The New York Times
39. Satchell C, Shanks G, Howard S, Murphy J (2011) Identity crisis: user perspectives on multiplicity and control in federated identity management. Behav Inf Technol 30(1):51–62
40. Tschorsch F, Scheuermann B (2016) Bitcoin and beyond: a technical survey on decentralized digital currencies. IEEE Commun Surv Tutorials 18(3):2084–2123
41. Vapen A, Carlsson N, Mahanti A, Shahmehri N (2016) A look at the third-party identity management landscape. IEEE Internet Comput 20(2):18–25
42. Zhang Y, Chen Z, Xue H, Wei T (2015) Fingerprints on mobile devices: abusing and leaking. In: Black Hat conference, Las Vegas, NV, USA

Index

A. Rattani et al. (eds.), *Selfie Biometrics*, Advances in Computer Vision
and Pattern Recognition, https://doi.org/10.1007/978-3-030-26972-2