JOHANNES KEPLER
UNIVERSITY LINZ

Submitted by
Rainhard Dieter Findling

Submitted at
Institute of Networks and Security (INS)

Supervisor and
First Examiner
Univ. Prof. Priv.-Doz. DI Dr. René Mayrhofer

Second Examiner
Prof. Priv.-Doz Dr. rer. nat. Stephan Sigg

Co-Supervisor
a.Univ. Prof. Dr. Josef Scharinger

September 2017

Unobtrusive Mutual Mobile Authentication with Biometrics and Mobile Device Motion

Doctoral Thesis
to obtain the academic degree of
Doktor der technischen Wissenschaften
in the Doctoral Program
Technische Wissenschaften

JOHANNES KEPLER
UNIVERSITY LINZ
Altenbergerstraße 69
4040 Linz, Österreich
www.jku.at
DVR 0093696

For the Future

ABSTRACT

Authentication is an integral part of protecting data on modern mobile devices from unauthorized physical access of third parties. However, it faces different challenges to suit users' needs. On the one hand classic authentication approaches like PIN or password are obtrusive especially on mobile devices. They impose cognitive load on users and their input on mobile devices is cumbersome due to small user interfaces and limited haptic feedback. This is further intensified by mobile devices being used more frequently but for shorter durations than classic computers. On the other hand biometrics can provide for less obtrusive authentication. However, disclosure of biometric data to third parties can have significant impact as they cannot be changed as easily as PINs or passwords. To avert this additional risk, embedded smart cards (SCs) can be used to process and store biometric data. As those are computationally limited this often leads to feature transformations and matching procedures also being limited. In addition, in contrast to users authenticating to mobile devices, devices usually do not authenticate to users. This enables hardware phishing attacks (users unwittingly authenticating to an identically looking but malicious phishing device).

This dissertation investigates unobtrusive mobile authentication for diverse situations in which authentication can be required. It thereby focuses on authentication approaches that utilize mobile biometrics and embedded sensors. We investigate generic biometric match-on-card (MOC) authentication that combines offline machine learning with simplification of features and authentication models to enable their usage on SCs. As the approach is generic it can be applied to different biometrics – demonstrated with gait and face biometrics – which can facilitate the transition of further mobile biometrics to using MOC techniques. We further investigate mobile token authentication to transfer the authentication state from an unlocked device (e.g. wristwatch) to a locked one (e.g. phone) by briefly shaking both devices conjointly. As shaking patterns are difficult to forge it is difficult for attackers to perform authentication when they do not have both devices under their control. We also investigate mobile device-to-user authentication as countermeasure to hardware phishing attacks and let devices communicate an authentication secret to users with vibration patterns. We evaluate our approach using publicly available data, which reveals authentication durations around 1-2 s and error rates between 0.2 and 0.02. This indicates both that our approach is feasible and that room remains for further improving unobtrusive mobile authentication, e.g. with additional approaches utilizing biometrics and sensors on mobile devices.

KURZFASSUNG

Mit modernen Mobilgeräten ist Benutzerauthentifizierung ein integraler Bestandteil zum Schutz von Daten auf Mobilgeräten vor unbefugtem, physikalischen Zugriff Dritter geworden. Herausforderungen mobiler Benutzerauthentifizierung umfassen kognitive Belastung (Merken von Geheimnissen), umständliche Eingabe (kleine Benutzerschnittstellen, wenig haptisches Rückmeldung) sowie eine höhere Nutzungsfrequenz von Mobilgeräten bei reduzierter Dauer pro Nutzung. Biometrien können hierbei verbesserte Anwendbarkeit ermöglichen, setzen Benutzer aber auch höheren Risiken hinsichtlich Diebstahl oder ungewollter Veröffentlichung von Biometrien aus. Integrierte Smartcards wiederum können zum Schutz von mobilen Biometrien verwendetet werden, gehen aber mit eingeschränkter Rechenleistung für Erkennungsverfahren einher. Des Weiteren authentifizieren sich Mobilgeräte üblicherweise nicht gegenüber ihren Benutzern, was sogenannte Hardware Phishing Attacks ermöglicht (bei diesen authentifizieren sich Benutzer unwissentlich gegenüber identisch aussehenden aber bösartigen Phishing-Geräten).

Die vorliegende Dissertation behandelt mobile Authentifizierungsverfahren für verschiedene Anwendungsszenarien unter sicherer Verwendung von Biometrien und integrierten Sensoren. Es wird eine generische, biometrische Authentifizierung unter Verwendung von Smartcards vorgestellt, welche Authentifizierungsmodelle und biometrische Eigenschaften vorab vereinfacht um deren Verwendung auf Smartcards zu ermöglichen. Des Weiteren wird ein Token-basierendes Authentifizierungsverfahren behandelt, welches durch kurzes gemeinsames Schütteln zweier Mobilgeräte den Authentifizierungszustand eines Geräts sicher auf das andere überträgt. Abschließend wird ein Verfahren zur Authentifizierung von Mobilgeräten gegenüber ihren Benutzern vorgestellt. Bei diesem übertragen Mobilgeräte unter Verwendung eines Vibrationscodes Informationen zu Benutzern – z.B. zeitgleich während sich Benutzer gegenüber dem Gerät authentifizieren. Alle Ansätze werden mit öffentlich verfügbaren Datensätzen evaluiert und zeigen Fehlerraten zwischen 0,2 und 0,02 sowie eine Dauer im Bereich von 1-2 sec auf. Diese Ergebnisse unterstreichen die grundlegende Anwendbarkeit der Ansätze und zeigen gleichzeitig den verbleibenden Spielraum für weitere zukünftige Verbesserungen mobiler Authentifizierungsverfahren auf – u.a. durch Verwendung mobiler Biometrien sowie integrierter Sensoren.

STATUTORY DECLARATION

I hereby declare that the thesis submitted is my own unaided work, that I have not used other than the sources indicated, and that all direct and indirect sources are acknowledged as references.
This printed thesis is identical with the electronic version submitted.

Linz, September 2017

Rainhard Dieter Findling

FOREWORD

Our modern world and everyday lives have become deeply interconnected with technology. Technologies are developed and invented increasingly faster with increasingly shorter long-term testing phases before their usage. As a result we rely on newer technologies less tested for long-term side effects on both global and individual level. Such technologies might introduce side effects that are revealed only long after whole societies have started using them. Those might not only be of technical nature, but could as well be of cultural, psychological, or ecological nature, to just name a few. Once a technology has established itself, societies and individuals might no longer be able to refrain from using it, even if it brings significant negative side effects or such are discovered later on. We see diverse examples already affecting our world as a whole as well as individuals on global scale, such as data privacy with the multitude of computers around us or the changes of global climate. With a globalized and interconnected humanity the question for societies and individuals in and after the 21. century concerning new inventions and technologies might consequently change from: "Do I really need to use technology X, given its advantages and drawbacks?" to "Can I *afford* to not use technology X (to avoid its drawbacks)?". This is because *not* using a technology that is considered standard in our modern world might bring other significant drawbacks for individuals. For example, people refraining from using such technologies might be confronted with disadvantages in terms of job choices, relationships, might need to spend additional time or money, and many more – which puts people under pressure to use such technologies anyway.

The impact new technologies might have should be taken into consideration during their development instead of neglecting or ignoring them. Thereby, the goal should not be to in general prevent development, inventions, or new technologies. It should instead be to see the prediction of long-term effects of new technologies as a requirement alongside development and to consider the gained knowledge to adapt development accordingly. To put it simple: it is better to steer proactively by design than to steer reactively by limitation of defect. The research area this thesis belongs to is itself only one small part of this wider context. While this thesis thereby is merely one tiny piece in the big picture, one tiny step on a long way, readers are invited to see it within this big picture and as one tiny step into what the author considers to be the right direction.

ACKNOWLEDGMENTS

First of all I want to thank René Mayrhofer for his extensive support over the past years. René manages to both show confidence in his students and their work, thus giving them enough free rein and time to experiment, discover, and comprehend the diverse aspects of their field of science – while at the same time properly communicating the need to take scientific responsibility for the research conducted. The excellent mixture of guidance, self-determination, experimentation, opportunities given, and forgiveness of mistakes is what I profited from most. Retrospective, this is, above all, what made the past years so precious. I further want to thank Josef Scharinger and Stephan Sigg for their valuable input and detailed feedback during conducting the research and writing the thesis. Additional thanks go to my colleagues including Daniel Hintze, Muhammad Muaaz, Michael Hölzl, Peter Riedl, Michael Roland, Kathrin Kefer, and Clemens Holzmann for the cooperation in the many different projects we were involved together, the great many discussions and brainstormings, and for all the interesting findings and fun we had over the past years. I also want to thank the staff of the University of Applied Sciences Upper Austria, Campus Hagenberg, and the Institute of Networks and Security at the Johannes Kepler University Linz for their steady support in both scientific and administrative issues over the past years.

Cordial thanks go to my family and friends, especially to my parents Helga and Johann Findling, to my brother Ronald Findling, and to Erika and Bernadette Peherstorfer. Your steady and continuous support, the opportunities you showed and gave me, the interesting discussions, the positive and necessary distractions, the backing in difficult situations, and especially the things we have seen and experienced together throughout all those years were and are essential for me to be determined and focus my vigor on work like this present thesis.

This work has partially been carried out within the scope of *u'smile*, the Josef Ressel Center for User-Friendly Secure Mobile Environments, funded by the Christian Doppler Gesellschaft, A1 Telekom Austria AG, Drei-Banken-EDV GmbH, LG Nexera Business Solutions AG, NXP Semiconductors Austria GmbH, and Österreichische Staatsdruckerei GmbH. Parts of this dissertation have previously been published or are submitted for review in [103–107, 229, 230].

CONTENTS

Part I

AUTHENTICATION IN MOBILE ENVIRONMENTS

1 INTRODUCTION

Personal mobile devices have become an integral part of our modern society. In recent years both the amount of devices as well as the amount of tasks that those devices are involved with in everyday life have risen significantly. This inevitably leads to those devices increasingly having access to, processing, and storing private and sensitive information. In case such information would be disclosed to unauthorized third parties it could be used maliciously in several ways. Well known examples would include surveillance, espionage, or blackmailing. As a result, data on modern mobile devices deserves adequate protection from disclosure.

For protecting data processed and stored on mobile devices one core question arises: what is the root of trust on mobile devices? Which parts of a mobile device ecosystem should be considered trustworthy and safe, and which require additional security measures? For instance, all components of a mobile device itself could be manipulated. This includes the mobile device hardware (CPU, sensors, etc.), bootloader, and operating system (including the kernel) as well as applications running on the device. Further, mobile devices could physically be accessed by third parties like they would be legitimate users in order to obtain access to processed and stored data. In addition, other parts of mobile environments might not be trustworthy either. This includes other devices, computers, and services a mobile device communicates with over networks. Well known examples would include cloud storage or swapped out processing to reduce computational requirements and battery consumption on mobile devices. Hence, to protect data on mobile devices a broad variety challenges arise [27, 226]. An important aspect of solutions to these is that they should not impede the everyday usage of mobile devices. Amongst others, the challenges towards protecting data on mobile devices include:

- How to build a root of trust (e.g. secure hardware) for mobile devices?
- Based on the root of trust, how to achieve operating system level security? This could include a chain of trust that goes from hardware over firmware and bootloader to the operating system in an effective way.
- How to achieve application level security? This could include third parties being unable or hardly able to use applications for

malicious purposes, limiting the impact of potentially malicious applications.

- How to communicate hardware and software based trust of devices to users? This could include indication of security factors to prevent e.g attacks based on deception of users.
- How to protect mobile devices from unauthorized physical access of third parties?
- How to incorporate external third party services in secure ways (e.g. external storage, swapped out computations)?

In this dissertation we focus on the challenge of how to protect data on mobile devices from disclosure by unauthorized physical access to the device. Overall, solutions addressing the above challenges need to use a corresponding threat model. For example, while preventing disclosure of mobile device data to national or state agencies will require a much broader and thorough threat model the same might be considerably simpler for less sophisticated attackers. National or state agencies might very well gain access to hardware manufacturers to inject malicious components during device manufacturing or inject vulnerabilities or backdoors in operating systems. Protection against such is considerably more difficult than against attackers with less capabilities who are unlikely to have such broad options for attacks. However, physically accessing mobile devices to obtain private information is not connected to any special skills and only requires attackers to gain physical access to the device by any means necessary. Consequently, such attacks could be carried out by nearly everyone in our modern society including family members and office colleagues as well as pickpockets in public transport or places. This is why protecting mobile devices from unauthorized physical access is an integral part of protecting data on mobile devices.

The most commonly used way of preventing unauthorized physical access to mobile devices is using authentication mechanisms and device locking functionality. Thereby legitimate users have to unlock mobile devices before using them by performing authentication while unauthorized users cannot unlock such devices as they fail authentication. Well known variants of mobile device authentication include PIN, graphical pattern, and biometrics like fingerprints. Although those approaches are the most commonly used ways of authentication on modern mobile devices they bring significant drawbacks with them. These include obtrusiveness in the form of additional cognitive load (users having to remember and recall a secret) as well as additional time to perform the authentication because input of secrets can be cumbersome due to small screens and limited haptic feedback. While biometrics do not bear cognitive load on users they deserve even higher protection than PIN or graphical pattern. This is because they cannot be chosen freely but are predefined and fixed per user and for all its applications, and because they cannot easily be changed in case they are disclosed to third parties. As a consequence, while the usage of biometrics eases mobile authentication it also exposes its users to additional risks. Despite their drawbacks those approaches (especially PIN and graphical pattern) are still the most frequently used authentication approaches with current mobile devices in 2017 [213]. In addition, usually a single authentication approach is employed, therefore is the same for all situations in which authentication is required. This makes suiting user needs in those naturally very diverse situations even more challenging.

These issues indicate that there is room for improvements towards less obtrusive mobile authentication approaches that better suit the diversity of situations in which mobile authentication is required. Within the greater goal of unobtrusive mobile device data protection, this dissertation therefore investigates new ways and alternative approaches to unobtrusive authentication with mobile devices, using biometrics and embedded sensors and without exposing user data to additional risks.

1.1 RESEARCH QUESTIONS

The work in this dissertation is organized around the following research questions:

- How can authentication that is employed to protect data on mobile devices from unauthorized physical access of third parties suit the large variety of situations in which authentication might be required?

- How could authentication with multiple mobile devices be used as advantage rather than a drawback?

- How to protect biometrics used for authentication on mobile devices from disclosure? How to apply such protection to multiple biometrics in order to aid secure usage of different biometrics on mobile devices in the future?

- How can mobile users be protected from hardware phishing attacks, that is them being deceived into unwittingly revealing sensitive information to identically looking but malicious phishing devices?

As a result, this dissertation investigates new ways of authentication with biometrics and sensors on mobile devices, demonstrates their feasibility, and evaluates their authentication performance with a respective threat model.

1.2 CONTRIBUTIONS

This work contributes to authentication in mobile environments in different ways. Subsequently we briefly highlight direct contributions (Sec. 1.2.1) as well as indirect or related contributions that – despite their relation – are not part of this dissertation (Sec. 1.2.2).

1.2.1 *Main Contributions*

1.2.1.1 *Generic Mobile Biometric Match-on-Card Authentication*

To protect biometrics used on mobile devices, a generic machine learning based biometric match-on-card authentication technique is developed (Cha. 5). The approach is generically applicable to different biometrics and different smart card architectures. It uses offline machine learning to generate and simplify an authentication model that can be used on smart cards without requiring retraining for enrolling users. Publications include:

- **Rainhard Dieter Findling**, Michael Hölzl, and René Mayrhofer: Mobile Match-on-Card Authentication Using Offline-Simplified Models with Gait and Face Biometrics. IEEE Transactions on Mobile Computing (TMC). Submitted for review.

- **Rainhard Dieter Findling**, Michael Hölzl, and René Mayrhofer: Mobile Gait Match-on-Card Authentication from Acceleration Data with Offline-Simplified Models. Proc. MoMM 2016: 14th International Conference on Advances in Mobile Computing and Multimedia, ACM, 2016, 250-260.

1.2.1.2 *ShakeUnlock: Transferring Authentication States Between Mobile Devices*

To provide for additional authentication possibilities on mobile devices, ShakeUnlock is developed (Cha. 6). ShakeUnlock transfers the authentication state of an already unlocked device (to which users already authenticated) to another, still locked device to unlock it. Users briefly shake both devices conjointly to perform ShakeUnlock which uses sensed acceleration of both devices to ensure they have actually been shaken by the same hand. Publications include:

- **Rainhard Dieter Findling**, Muhammad Muaaz, Daniel Hintze, and René Mayrhofer: ShakeUnlock: Securely Transfer Authentication States Between Mobile Devices. IEEE Transactions on Mobile Computing (TMC), 2017, 16, 1163-1175.

- **Rainhard Dieter Findling**, Muhammad Muaaz, Daniel Hintze, and René Mayrhofer: ShakeUnlock: Securely Unlock Mobile Devices by Shaking them Together. Proc. MoMM 2014: 12th International Conference on Advances in Mobile Computing and Multimedia, ACM, 2014, 165-174. *Best paper award MoMM 2014.*

- René Mayrhofer, Helmut Hlavacs, and **Rainhard Dieter Findling**: Optimal Derotation of Shared Acceleration Time Series by Determining Relative Spatial Alignment. International Journal of Pervasive Computing and Communications (IJPCC), 2015, 11, 454-466.
- René Mayrhofer, Helmut Hlavacs, and **Rainhard Dieter Findling**: Optimal Derotation of Shared Acceleration Time Series by Determining Relative Spatial Alignment. Proc. iiWAS 2014: 16th International Conference on Information Integration and Web-based Applications and Services, ACM, 2014, 71-78. *Best paper award iiWAS 2014.*

1.2.1.3 *Device-to-User Authentication Using Vibration Patterns*

As first step towards protecting mobile device users from hardware phishing attacks, mobile device-to-user authentication is investigated (Cha. 7). In a first approach it uses device vibration to communicate authentication information from devices to their users. Publications include:

- **Rainhard Dieter Findling** and René Mayrhofer: Towards Device-to-User Authentication: Protecting Against Phishing Hardware by Ensuring Mobile Device Authenticity using Vibration Patterns. 14th International Conference on Mobile and Ubiquitous Multimedia (MUM '15), ACM, 2015, 131-136.

1.2.2 *Other Contributions*

Besides the main contributions of this dissertation the author has made substantial contributions to other work that, despite its close relation the main contributions, is not part of this dissertation.

1.2.2.1 *Mobile Device Usage Characteristics*

Usage behavior of modern mobile devices is investigated on a large scale in an mobile device usage analysis. Special focus lies on usage of mobile devices while being locked as well as differences by context. Publications include:

- Daniel Hintze, Philipp Hintze, **Rainhard Dieter Findling**, and René Mayrhofer: A Large-Scale, Long-Term Analysis of Mobile Device Usage Characteristics. Proc. ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 2017, 1. In print.

- Daniel Hintze, **Rainhard Dieter Findling**, Sebastian Scholz, and René Mayrhofer: Mobile Device Usage Characteristics: The Effect of Context and Form Factor on Locked and Unlocked Usage. Proc. MoMM 2014: 12th International Conference on Advances in Mobile Computing and Multimedia, ACM Press, 2014, 105-114.
- Daniel Hintze, **Rainhard Dieter Findling**, Muhammad Muaaz, Sebastian Scholz, and René Mayrhofer: Diversity in Locked and Unlocked Mobile Device Usage. Proc. 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication (UbiComp '14), ACM Press, 2014, 379-384. *Winner of the UbiComp/ISWC 2014 Programming Competition award.*

1.2.2.2 *CORMORANT: Framework for Multi-Modal Mobile Authentication*

To combine different authentication modalities on mobile device CORMORANT has been developed. CORMORANT is an authentication framework that enables combination of different authentication modalities on mobile devices regardless of their attributes. It thereby accounts for transparently collecting individual authentication results and deriving an overall authentication decision from them. This enables developers of novel mobile authentication approaches to focus on the authentication approach itself and leave the usage of its authentication result to CORMORANT. Publications include:

- Daniel Hintze, Muhammad Muaaz, **Rainhard Dieter Findling**, Sebastian Scholz, Eckhart Koch, and René Mayrhofer: Confidence and Risk Estimation Plugins for Multi-Modal Authentication on Mobile Devices using CORMORANT. Proc. MoMM 2015: 13th International Conference on Advances in Mobile Computing and Multimedia, ACM, 2015, 384-388.
- Daniel Hintze, **Rainhard Dieter Findling**, Muhammad Muaaz, Eckart Koch, and René Mayrhofer: CORMORANT: Towards Continuous Risk-Aware Multi-Modal Cross-Device Authentication. Proc. 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication (UbiComp '15), ACM, 2015, 169-172.

1.2.2.3 *Mobile Wrist Vein Authentication*

In order to integrate other biometrics in mobile authentication, mobile wrist vein authentication has been investigated. For users wearing wrist watches wrist veins bear the advantage of being right below the watch. This could enable completely unobtrusive and transparent wrist vein authentication for modern wrist watches in the future. Publications include:

- Pol Fernández Clotet and **Rainhard Dieter Findling**: Mobile Wrist Vein Authentication Using SIFT Features. Proc. Eurocast 2017, Springer, 2017. In print.

1.3 THESIS ORGANIZATION

This dissertation is organized in two parts: part I focuses on the background, related work, and the current state of the art, while part II contains the contribution, including concepts and evaluation data as well as evaluation results and findings. In part I we describe modern mobile environments as well as the problem of using classic authentication in such (Cha. 2). We further highlight approaches to improve authentication with mobile devices, including the discussion of details of different knowledge, biometrics, and token-based authentication approaches, as well as the concept of devices also authenticating to users (Cha. 3).

In part II we introduce our contribution by giving an overview of our approach (Cha. 4). We investigate generic mobile biometric match-on-card authentication in depth, stating details about the threat model, technical approach and solution, evaluation data and setup, evaluation results, as well as corresponding findings (Cha. 5). We investigate ShakeUnlock in depth, again stating the threat model, technical approach and solution including details on its constituent parts, evaluation data and data recording, evaluation setup, as well as results and findings (Cha. 6). We further investigate device-to-user authentication and the usage of device vibration patterns to communicate information from mobile devices to their users, including the vibration code design and its evaluation, results, and findings (Cha. 7). In addition, we recap how our approach changes the overall threat model for unauthorized physical third party access to mobile devices. We thereby focus on how our approach impedes possible attacks as well as which threats remain or have newly arisen with our approach (Cha. 8). Lastly, we conclude and give an outlook for future work (Cha. 9).

2 SECURITY AND AUTHENTICATION WITH RESPECT TO MOBILE ENVIRONMENTS

This chapter discusses the integration of mobile devices into modern everyday life and the data mobile devices thereby get access to (Sec. 2.1), the importance of protecting this data from unauthorized third party access as well as possibilities for such access (Sec. 2.2), and the drawbacks of applying classic authentication mechanisms from desktop computers on mobile devices (Sec. 2.3).

2.1 THE MOBILE DEVICE ECOSYSTEM: NOW AND IN THE FUTURE

Portable computers in the form of mobile devices have become an important part of modern life. Mark Weiser envisioned in 1991 [360] that computers will become increasingly smaller, ubiquitous, and fade into the background, while at the same time becoming computationally more powerful. This vision has become reality over the past 25 years, as nowadays examples for such devices include – but are not limited to – smart phones, smart watches, or tablets, which have more computational capabilities than many mainframe computers in the past century. One core difference of such devices to classic computers is their mobility and continuous availability. In contrast to classic computers, which are less mobile or stationary and require additional time before usage when they are started, mobile devices are with their owners in many cases and are usually turned on throughout the day. Consequently, they are available to users most of the time and ready to be used. Another core difference between modern mobile devices and classic computers is the increased sensing and connectivity capabilities. These allow cooperation amongst devices and services with frequent information gathering and exchange. Examples include devices sensing their contexts or environments using embedded sensors, such as accelerometers, gyroscopes, magnetometers, temperature and proximity sensors, cameras, or microphones.

The continuous availability of modern mobile devices and the vast amount of information available to them enables them to ubiquitously and invisibly aid users in many different situations throughout daily life. This can be done in both solving small everyday tasks more cleverly than existing approaches, or in providing aid for tasks for which no aid existed previously. An example for the latter would include mobile devices that monitor a user's sleep and derive the sleep cycles – which can consequently be used to wake users when it is easiest to wake up within certain boundaries. As a result mobile devices aid their users by e.g. saving time, money, help in organizing things and reduce cognitive load, or help in keeping in touch with work, friends, and family. Such aid is not limited to smart phones, smart watches, or tablets, but could also be offered e.g. by intelligent and connected cars, mobile devices in the areas of sport, health, and medicine, or devices and sensors used with home automation. The easier the usage of mobile devices, the more they are available in everyday tasks, the more information is available to them, and the broader the application possibilities for such devices become. This consequently leads to devices being used in more and more situations and for more and more different tasks.

As a result, users have started to rely and become dependent on their mobile devices. Well known examples include telephony, messaging, or information lookup while being on the move. Hence, using mobile devices for everyday tasks is not only convenient but has virtually become a requirement to perform certain tasks in everyday life effectively. Thereby, the less devices can be left out, the more they are necessarily involved in such tasks, and the more information they will again get to sense, process, store, and exchange about their users.

In accordance to Mark Weiser's vision, in the future there might be even more small, unobtrusive, and connected mobile devices. However, more importantly, mobile devices will be more deeply integrated into everyday tasks than nowadays. Technical limitations of mobile devices will restrict their usage in less and less situations. This will lead to them being integrated to or required for everyday tasks more frequently, further increasing the dependency of their users on them, and consequently leading to them sensing, processing, storing, and exchanging even more information about their context and users. When extrapolating this vision to a more distant future and different devices, whole populations, countries, and economies could depend on their mobile devices for everyday tasks – ranging from tasks in the business sector to tasks in private life. Mobile devices in their current form are likely only one of the many aspects that participate to this process. Other devices from other areas, such as the currently intensively investigated area of automotive computing will play an important role in aiding users in everyday tasks too (e.g. intelligent systems built into autonomous cars). These systems will again increase the amount of data that is gathered, processed, stored, and exchanged about their contexts – as well as their users.

2.2 WHY MOBILE DEVICE DATA NEEDS TO BE PROTECTED

2.2.1 *Impact of Sensitive Data from Mobile Devices Being Disclosed*

The information sensed, gathered, processed, stored, and exchanged on modern mobile devices should be considered private and be protected from unauthorized access of third parties. Even nowadays mobile devices are able to access a fairly comprehensive amount of data about their context and their users [149–151, 329]. Well-known examples include, but are not limited to, communications (email, SMS, instant messaging), context information (location), access to non-public networks (WiFi, VPN), access to payment or identity management applications, photos, documents, and even health related information (e.g. heart rate). In addition, with the "Bring your own device" trend (cf. [237, 337]), employees start to store and process business and company related data on private mobile devices. This information should in general be treated as private and sensitive and therefore should be protected accordingly from unauthorized access of third parties.

Giving some simple examples for which purposes third parties could maliciously use such information in case it is not protected accordingly: mobile devices could be used for undetected surveillance of individuals or to disclose private or sensitive information processed by or stored on those devices [245]. The latter could be used to perform espionage (e.g. industries or politics) or for blackmailing individuals or companies, to only name two examples. Thinking ahead, through the data mobile devices process and store they hold a part of users' identities. By obtaining control over a mobile device, attackers could potentially obtain partial control over a user's identity. This could enable identity hijacking, which enables attackers to block, delete, or alter a user's identity. Attackers could further conduct malicious actions in the name of the user. Both can have severe consequences for affected individuals.

The potential harm of such data falling into the hands of unauthorized third parties increases alongside the amount and quality of the data processed and stored on mobile devices. Consequently, the more data mobile devices are able to sense, aggregate, process, and store, the higher the chances that disclosed data can be useful for attackers, hence the bigger the potential resulting impact for users. This is why mobile device security becomes more important with increasing amounts of information being available to mobile devices.

2.2.2 *Threat Model Overview: Unauthorized Access to Mobile Data*

As attackers could obtain unauthorized access to mobile device data on different ways the overall threat model can be divided into several layers ranging from hardware to software aspects. In terms of hardware, control over any critical hardware in mobile devices (including CPU and other integrated circuits with access to CPU and/or device memory) would enable attackers to obtain full control over devices. This would include access to data stored and processed by applications running on such devices. An exemplary attack would include manipulation of blueprints before device manufacturing or manipulation of manufactured devices before they reach customers. Other ways of accessing mobile device data with manipulated hardware would include e.g. sensors with malicious functionality which would give attackers access to sensor data.

In terms of software, attacks could be performed e.g. on the bootloader, operating system, application, or even user interface (UI) level. Exemplary attacks on bootloader or operating system include exploitation of vulnerabilities on mobile devices (e.g. privilege escalation) that allow attackers to manipulate and obtain control over bootloader and/or operating system. In terms of operating systems, such attacks could involve vulnerabilities in either the operating itself (including the kernel) or included third party libraries. Such attacks could be carried out either using physical access to devices, using non-privileged software installed on devices, or remotely using security flaws in network related operations. A prominent example of the latter with third party libraries includes CVE-2015-7547, which allows attackers to potentially perform remote code execution using the libc library that is shipped with most modern operating systems, thereby effectively demonstrates the potential impact of such flaws[1]. Again, obtaining control over the bootloader or operating system would enable attackers to observe/manipulate any application spawned atop. Exploitation of vulnerabilities of individual applications can again lead to privilege escalation – if the application is executed with elevated privileges – but can at least be used to obtain access to data processed and stored by the affected application. Other attacks on application level include users unwittingly installing and using malicious applications (e.g. Trojans) or benign applications relying on third party libraries with embedded malicious functions. Again, these could enable attackers to access data processed and stored by affected applications. Atop the mentioned attacks, user deception could be used to obtain access to mobile device data. For example mobile applications could perform phishing techniques to deceive users into entering sensitive information in malicious applications – while they believe to be using the correct application [285].

Beside the mentioned possibilities for unauthorized access to mobile device data on software and hardware level, physical access to mobile devices could be used to by attackers to access processed and stored data. Instead of exploiting or injecting vulnerabilities into soft- or hardware, attackers would use mobile devices like legitimate users. The device would thereby provide the same functionality to attackers as to users by providing access to private and sensitive information processed and/or stored on the device. In this work we focus on this issue: the unauthorized access of third parties to data processed and stored on mobile devices using physical access. Attacks based on

[1] CVE-2015-7547: `https://cve.mitre.org/cgi-bin/cvename.cgi?name=cve-2015-7547`

exploitation of vulnerabilities of soft- and hardware components of mobile devices are outside the scope of this work if not stated otherwise.

There are two major differences between attacks using vulnerabilities and physical access to mobile devices. While the first could be used by one attacker to obtain access to many devices, potentially even remotely, physical access requires proximity to the device, hence cannot be performed remotely in any way or on many devices in parallel. However, exploiting vulnerabilities or foisting malicious applications to users is connected to certain effort and requires certain knowledge about the exploitation process. In contrast, mobile devices could be physically accessed by attackers without requiring particularly uncommon knowledge about the device and only require attackers to obtain physical access to the device by any means. Consequently, attackers are not restricted to parties with sufficient knowledge and resources but could include family members, colleagues at work, passengers on streets and public transportation, and many more. While the mobility of mobile devices provides for convenience it also makes them easier to be forgotten, lost, or stolen than classic computers. This further lowers the effort for physical third party access to data on mobile devices. For example, while physically accessing data on a home computer might require burglary, a mobile device could be accessed or stolen e.g. in public transport in an instant if the owner is inattentive. Furthermore, as mobile devices are powered most of the time data could be accessed more quickly than with classic computers (which attackers might need to turn on before). This makes performing quick attacks without users noticing easier for mobile devices than for classic computers.

To summarize, information processed and stored on mobile devices needs to be protected accordingly from third party access. While attackers can use multiple ways to access this data our work focuses explicitly on preventing unauthorized physical access to mobile devices by third parties. To prevent unauthorized physical access authentication mechanisms can be employed – which we focus on in the next section.

2.3 CLASSIC AUTHENTICATION AND ITS IMPLICATIONS IN MOBILE ENVIRONMENTS

Unauthorized physical access to computer devices can be prevented using authentication: legitimate users can authenticate and use the device while other users cannot do so. For this purpose computers (including mobile devices) usually feature a locking mechanism that can lock the device and keep it locked while it is not actively used. Legitimate users need to unlock the device before usage by authenticating to it. Authentication can be categorized into 3 types: using knowledge, inherence (biometrics), and possession (tokens) [2, 124]. The most widely used type of authentication uses knowledge, e.g. by requiring users to enter a secret password or PIN. We therefore discuss the advantages and drawbacks of knowledge based authentication in general as well as with special regard to the mobile environment. Other authentication suitable for mobile devices, including graphical patterns and biometrics, are discussed in Cha. 3 as they can be seen as attempts to improve mobile authentication in terms of obtrusiveness.

2.3.1 *Knowledge Based Authentication: PIN and Password*

With knowledge based authentication users authenticate to devices using a secret only they know. Usually the secret is pre-shared between users and devices and users reveal this secrets to devices e.g. by entering it on a keyboard or keypad on or connected to the device. The most widely known and used knowledge based authentication approaches are PIN (numeric) and password (allowing a wide range of characters). They are used widely and in many fields of application, ranging from ATM machines and credit cards to authentication pads to open door and garages; or from logins to computers and network like WiFi or virtual private networks (VPN) to all kinds of Internet services like websites or mail services.

The theoretical authentication strength/security can be quantified as entropy[2]. For a certain approach and configuration the entropy S in bits is derived from the theoretically maximum possible amount A of different secrets that can exist [312] (Eq. 1).

$$S = \frac{\log(A)}{\log(2)} \tag{1}$$

The maximum possible amount A of different secrets thereby depends on the alphabet and length of the PIN or password [256]. For example, for a 4-digit numeric PIN with alphabet [0-9] $A = 10^4 = 10000$ possible secrets exist, which corresponds to an entropy of $S \simeq 13.2\,\text{bit}$. For a 8-character alphanumeric password with alphabet [a-zA-Z0-9] $A = 62^8 \simeq 2.18 \cdot 10^{14}$ possible secrets exist, which corresponds to an entropy of $S \simeq 47.6\,\text{bit}$. The higher this entropy the less likely brute force attacks are able to guess the one used secret from the full space of secrets. It is important to note that user chosen secrets are usually unevenly distributed in the full space of secrets, resulting in the real entropy of such secrets being lower than the theoretically possible one [256]. This issue is discussed in more detail in the next section.

[2] It is important to note that entropy is only one important factor in authentication. Attackers could use other weaknesses than a small entropy to attack an authentication approach.

2.3.2 *Knowledge Based Authentication: Advantages and Drawbacks*

From a user's perspective knowledge based secrets have certain advantages and drawbacks in comparison to other forms of authentication. Knowledge based authentication approaches are often purely software based, in contrast to e.g. biometrics or token-based authentication, which often require additional hardware [94]. This allows their implementation and usage on many different devices as they only need to feature some capability for users to enter their secret, which exists with many user interfaces. Further, knowledge based secrets can easily be exchanged with a new secret in case they become known to third parties or possible attackers.

In terms of drawbacks there usually exists a trade-off between security and usability: increasing security usually lowers usability and vice versa. In terms of authentication this means that not employing any authentication leads to the least obtrusive user experience while adding authentication or increasing its strength will lead to more obtrusiveness. This is well known for classic computers and mobile devices alike [3, 36, 136, 143]. With knowledge based authentication reasons for this include, amongst others, time required to perform the authentication and cognitive load imposed on users by the authentication secret. The effect of the first is noticeably stronger on mobile devices: the input of secrets on mobile devices is usually more cumbersome than on keyboards of classic computers. This is mainly caused by users being required to use virtual keyboards on small screens and less haptic feedback [21, 350]. As users want access to their devices as fast as possible this leads to certain users not employing knowledge based authentication at all [245]. Further, entering secrets on mobile devices being more cumbersome also leads to increased authentication failure rates [135]. This again increases the average time authentication takes on mobile devices.

Besides requiring additional time, knowledge based authentication also necessarily imposes cognitive load on users. This is the result of users being required to memorize and recall the authentication secret. This leads to a well known decrease in usability of PINs and passwords when using many or complex authentication secrets (cf. [3, 27, 36, 68, 72, 143, 300, 381, 384]). As a result, users tend to either not use authentication at all or to choose weak passwords that can more easily be memorized, but which are also more easily to predict or guess by attackers. This effect is even worse when users are required to memorize and maintain multiple different passwords which all increase the resulting cognitive load [15, 37, 256, 270, 317]. For example, Zhang-Kennedy et al. [381] analyze typical password rules, like not being allowed to reuse passwords, required password length and complexity, requirements to frequent changes of passwords, etc. They find that these lead to significant drawbacks for users. If passwords are required to be used they suggest to use strong passwords that are memorable (e.g. from mnemonic phrases – which are known to introduce weaknesses themselves [189]), to change passwords only if there is a reason to do so, to strategically reuse passwords, and to write down passwords but to protect them well (e.g. offline password list that cannot be obtained by attackers with obtaining control over a computer). These suggestions all have in common that they aim to lower the cognitive load imposed on users by knowledge based authentication approaches.

From an attacker's perspective knowledge based authentication can also be attacked using shoulder surfing [304, 334]. Knowledge based unlocking approaches are vulnerable to shoulder surfing attacks, whereat attackers watch the authentication process and thereby observe the authentication secret. One well known example of shoulder surfing would be attackers observing the PIN authentication on another user's mobile device which enables them to unlock the device once they obtain control over it.

Summarizing, knowledge based authentication has even stronger drawbacks on mobile devices than on classic computers. Besides the mentioned reasons this is further amplified by a higher usage frequency but shorter duration per usage of mobile devices [23, 140, 151, 161] – as well as a potential multitude of mobile devices requiring authentication. This causes an increased authentication-to-usage duration ratio on mobile devices, thereby an increased authentication overhead over usage time. As a result, this leads to knowledge based authentication being more obtrusive on mobile devices than on classic computers in general. However, PINs, passwords, and graphical patterns[3] are still the most widely used forms of authentication on modern mobile devices in 2017 [213]. This indicates that despite their drawbacks no other approaches have been able to provide for mobile users' needs or act as replacement yet. This further underlines the need for additional and alternative ways of authentication on mobile devices towards achieving less obtrusive authentication. Examples for approaches that aim to advance mobile authentication towards these goals are discussed in the next chapter.

[3] Graphical patterns are discussed in Sec. 3.1.2 as they are a special form of graphical passwords which have explicitly been developed to address drawbacks resulting from cognitive load of PINs and passwords.

3 APPROACHES TO IMPROVE AUTHENTICATION WITH RESPECT TO MOBILE ENVIRONMENTS

As discussed in the previous chapter, classic knowledge based authentication approaches such as using PINs or passwords bring significant drawbacks with them by being obtrusive for users. The obtrusiveness is further intensified when those are employed with mobile devices. To address those drawbacks, diverse authentication approaches have been investigated in previous research. While many of them have been designed to be employed with classic desktop computers, most of their advantages and drawbacks as well as their findings and takeaways apply to the mobile environment as well. In this chapter we give an overview of authentication approaches that strive for being unobtrusive, hence to facilitate authentication actually being used instead of being rejected due to its drawbacks [9]. These cover knowledge based authentication approaches, including graphical passwords and patterns, different biometrics, token-based authentication, as well as the advantages of combining different authentication modalities. We discuss these aspects with special regards to the mobile domain. We further highlight how biometrics can be protected from disclosure to unauthorized third parties – as this is an aspect of paramount importance when biometric authentication is employed.

3.1 KNOWLEDGE BASED AUTHENTICATION

An important issue of knowledge based authentication including PINs or passwords is that it bears cognitive load on users. To address this issue, other knowledge based authentication approaches have been investigated in the past that aim at achieving reduced cognitive load. In this section we discuss graphical passwords as one important example of such approaches as well as graphical patterns as a special case that is frequently employed to unlock current mobile devices.

3.1.1 *Graphical Passwords*

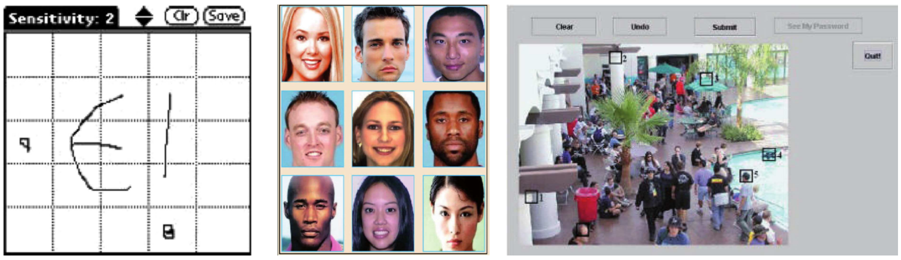
Humans are better at memorizing and recalling visual information than characters or numbers like PINS or passwords [326]. Graphical passwords are based on the following idea: they represent passwords in visual or graphical form to increase their memorability for users, hence to reduce cognitive load. There exist three major types of graphical passwords: pure recall based, recognition based, or cued-recall based graphical passwords [132]. In the following we discuss some important concepts behind graphical passwords and highlight their relation impact on authentication in the mobile domain. For a more comprehensive review of graphical passwords in general we refer to the surveys of Bidde et al. [31], Hafiz et al. [132], and Suo et al. [326].

3.1.1.1 *Graphical Passwords Based on Recall, Recognition, and Cued-Recall*

Pure recall based graphical passwords require users to memorize their secret without showing any information to help recalling the password. Such would e.g. be drawing password on a blank screen or on a grid displayed on the screen – without using any possibly helping background image. Classic passwords can be considered to be pure recall based. Examples for pure recall based graphical passwords include draw a secret (DAS) [172] where users draw their secret on a 2D grid using either a computer mouse or pen (Fig. 1a). The secret is represented as the ordered sequence of cell coordinates, which essentially is the order in which users connect cells. DAS serves as basis for a number of subsequent approaches. The usage of background images with DAS has been proposed with BDAS [95]. Users choose the background image they want to draw a secret on. Though this leads to more complex secrets being chosen than with DAS users still choose weak secrets [31]. In contrast to DAS, PassDoodle [122, 351] utilizes a completely freehand drawing as password. This requires a more complex matching of secrets to derive two drawings being the same than with DAS. MasterDoodle [126] in extension of PassDoodle specifically designed for managing many different such passwords. With PassShapes [362] passwords consist of strokes. There exist a total of 8 strokes, each represented by a simple line stroke into a certain direction and covering a total angle of 45°. A sequence of such strokes thereby represents a password. Thereby each character can be drawn in any size or on any position on the screen to be recognized correctly. Another variant of DAS is PassGo [333] which uses a different grid and different cells to avoid DAS secrets being close to cell borders easily causing authentication failure due to touching a wrong cell.

Recognition based graphical passwords require users to recognize and select one piece amongst many displayed pieces. This could be e.g. by the recognition and selection of familiar faces or familiar icons amongst many faces or icons displayed on a screen. Determining the correct piece of the secret is thereby based on recognition of displayed information instead of pure recall. Examples for recognition based graphical passwords include PassFaces[1] [45, 84, 334] and (Pass)Story [84]. With PassFaces, users recognize and select a familiar face from usually 9 displayed faces (Fig. 1b). There are multiple

[1] PassFaces online presence: http://passfaces.com/



(a) Draw a secret (DAS) (b) PassFaces (c) PassPoints

Figure 1: Examples for graphical passwords with (a) recall based draw a secret (DAS), (b) recognition based PassFaces, and (c) cued-recall based PassPoints [31].

rounds of this selection in which users have to select the correct face each time to successfully authenticate. As a variant of PassFaces, images displayed by (Pass)Story show a thematic context instead of a face. A series of such thematic contexts thereby represents a "story". Users select images with context according to their story during authentication, thereby the story acts as the authentication secret. One advantage of (Pass)Story is that multiple images can be of the same context, thereby the displayed images for entering the same story can be different each time.

Cued-recall based graphical passwords combine aspects from pure recall and recognition based graphical passwords. They usually aid users in recalling and entering their secret to the mobile device by displaying information related to the secret that is recognized by users. This could be done using e.g. a background image on which users utilize keypoints to then draw their secret onto the image. Both memorability and input of the recalled secret is thereby aided by the displayed visual information. Examples of cued-recall based graphical passwords include the patent of Blonder [34] in which a password is a series of clicks onto predefined points in a displayed image. A widely considered example is PassPoints [365, 366] which is based on the approach by Blonder but uses target areas instead of points in the image which can be clicked with certain tolerance (Fig. 1c). Variants of PassPoints exist, such as Cued Click Points [65], where instead of a single image a series of images in presented. The subsequent image thereby depends on the click area on the previous image. Another example includes persuasive cued click points [63] which additionally tries to influence users towards selecting better passwords containing more entropy.

3.1.1.2 *Graphical Passwords with Mobile Devices*

Though many approaches towards graphical passwords can be used on both classic computers and mobile devices, few approaches have been designed specifically with mobile devices in focus. For example, Jansen et al. [169–171] were amongst the first to explicitly target mobile devices with graphical passwords. During enrollment, users select a theme such as "sea" or "cat". They then get presented either one image related to the chosen theme with a grid overlay or a grid containing smaller images (icons) where some are related to the chosen theme. Users then select tiles as password, where the password consists of both tile content and order of tiles. During authentication the according images need to be selected in correct order. Due to the amount of tiles being restricted to 30 the resulting password space is considered small. SecureUnlock [306] combines different authentication approaches for mobile devices using Android, including NFC tags and GesturePuzzle. GesturePuzzle is a recognition based graphical password that aims to be less affected by shoulder surfing. With it, different symbols are presented to users aligned in a grid. Users consider a predefined subarea of the grid that indicates the action that should be performed, such as "draw a square around other icons". The user then performs the indicated action on any displayed symbols to perform authentication. The authors estimate a duration of about 5-8 s to perform the proposed authentication.

Other approaches include Chang et al. [58], who combine keystroke dynamics based on time and pressure features with graphical passwords to enhance authentication security on mobile devices. They evaluate their approach to result in 12.2% EER without and 6.9% EER with using pressure features. Chiang et al. [61] propose touchscreen multi layered drawing (TMD), which is a recall based graphical password. TMD uses large detached cells to reduce accuracy errors with users. They evaluate their approach to result in 86-100% authentication success rate with 15-18 s authentication duration. The related approach of Sabzevar et al. [295] does not directly target mobile devices but utilizes them in the authentication process. They propose to combine aspects of recognition and recall based graphical passwords with mobile devices as a token for two-factor authentication. Thereby the mobile device is required as second device answer a graphical password based challenge. This enables users to enter a password on untrusted terminals. Further, lost or stolen device do not pose immediate danger in terms of fraud authentication.

Summarizing, graphical passwords seem to be easier to memorize and cause less errors during authentication than classic passwords [45, 98, 326, 334, 367] (with different studies indicating authentication success from under 50% up to 100% [31] and theoretically possible password entropy in the range of 4.5 bits [94] to 300 bits [31], depending on the study setup). But graphical passwords still suffer from the exact same drawbacks as classic passwords. As with all user chosen knowledge based authentication secrets, users have difficulties remembering their graphic passwords [62, 98] and therefore show a tendency to choose simple graphical passwords that can more easily be memorized, but which are also more easy to attack [62–64, 89, 98, 258, 338–340]. An exemplary study highlighting the extent of this problem would be [84], where users showed tendencies to choose passwords related to personal attributes such as race or gender, and where about 10% of male PassFace passwords could have be guessed by mere 2 guesses with personal attributes know to attackers. Additionally, due to easier memorability of graphical passwords some approaches also seem be more easily attackable with shoulder surfing [334]. In contrast, other studies report an average of 7.5 required observations by attackers to perform shoulder surfing for high entropy approaches (small images, low image quality) and 4.5 observations with low entropy approaches [94]. Further, using a large amount of small icons can be problematic on mobile device screens as they tend to be significantly smaller than screens of classic computers. Entering a secret might become especially cumbersome in such cases (cf. [367]).

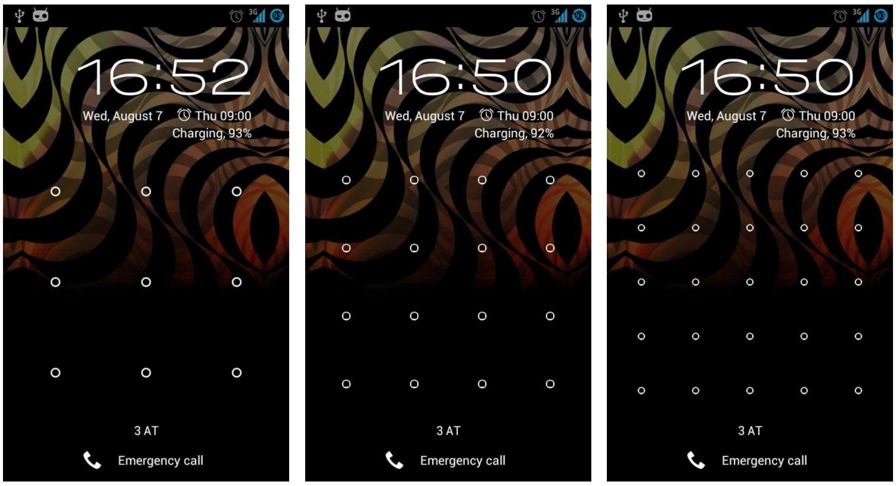
Besides all the mentioned advantages and drawbacks of graphical passwords, the remaining and most severe drawback is authentication duration. Most approaches report an authentication duration much longer than using classic PIN or password, ranging from 5 s to over 90 s with the majority of approaches in the range of 10 s to 20 s [31, 45, 94, 367]. This can be seen as severe drawback for mobile users accustomed to shorter authentication durations and would be a possible explanation for the small adoption of graphical passwords with mobile devices. Therefore, with their advantages and drawbacks, graphical passwords can only be assumed suitable for authentication on mobile devices in some situations. They should therefore not be seen as possible full replacement of classic passwords on mobile devices, but more as complementary option for mobile authentication [305].

3.1.2 *Graphical Pattern*

Graphical pattern unlock is a special form of graphical password specifically designed for authentication on modern mobile devices. As with other graphical passwords the overall goal of graphical patterns is to be less obtrusive than PIN or password based authentication by being easier to memorize and recall, consequently to bear less cognitive load on users.

3.1.2.1 *Functional Principle of Graphical Patterns*

The functional principle behind graphical pattern unlock is to connect dots displayed on the mobile device screen with the finger in the correct order. The authentication secret thereby is which dots have to be connected in which order. A grid of $3 \times 3 = 9$ dots is most frequently used (Fig. 2), but other grid sizes and other non-grid arrangements of dots are possible as well.



(a) $N = 9$ dots (b) $N = 16$ dots (c) $N = 25$ dots

Figure 2: Graphical pattern authentication using different amounts of connectible dots in a grid arrangement [102].

The theoretically possible size of the set A of passwords for a graphical pattern where dots can be connected once in arbitrary order depends on the amount of dots N and the minimum N_{min} and maximum of N_{max} dots a password can consist of (Eq. 2) [102]. From this the resulting maximum possible entropy S of this graphical pattern can be derived (Eq. 3) [312].

$$A = \sum_{n=N_{min}}^{N_{max}} \frac{N!}{(N-n)!} \tag{2}$$

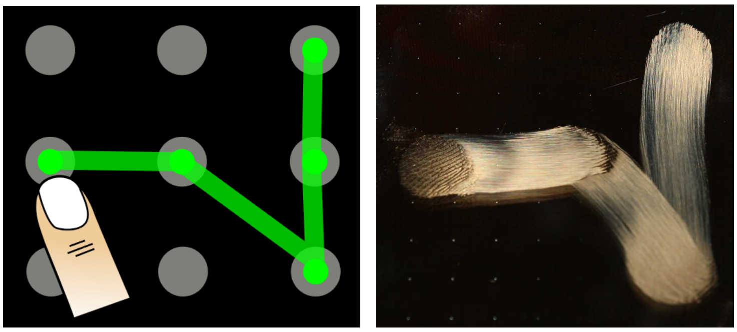
$$S = \frac{\log(A)}{\log(2)} \tag{3}$$

With the frequently used grid of $N = 3 \times 3 = 9$ dots, $N_{min} = 1$ and $N_{max} = 9$, this would result in 986409 possible passwords which corresponds to an entropy of about 19.91 bits [102].

Though mobile users have shown some acceptance of graphical pattern as device unlock on the Android platform [149–151], graphical pattern authentication is still a form of graphical password and thereby brings the same advantages and drawbacks as other graphical passwords. Like PIN, passwords, other graphical passwords, or any other knowledge based authentication approaches, graphical patterns bear some cognitive load on users that – while being easier to memorize and recall due to it being a graphical password approach – cannot be canceled out completely. Users still face the issue of having problems to remember more complex patterns or multiple patterns for multiple devices and authentication services. Consequently, users also tend to choose weak graphical patterns for unlocking their mobile devices and tend to reuse patterns across multiple devices [379]. Additionally, in terms of duration on mobile devices drawing a graphical pattern can be considered to be a little more obtrusive than entering a PIN or password. Unlocking usually takes a little longer with patterns (on average 2.7 s [150] to 3.1 s [377]) than entering a PIN or password (on average 1.5 s [377] for PINs and 2.5 s [150] for PINs and passwords combined). Furthermore, as with most knowledge based secrets, drawing a graphical patterns also requires user attention as users most likely have to look at the screen while performing the authentication.

3.1.2.2 *Shoulder Surfing and Smudge Attacks on Graphical Patterns*

Besides those user-centric drawbacks, graphical patterns can be attacked by both shoulder surfing and smudge attacks. With graphical patterns, shoulder surfing works the same way as with all other knowledge based secrets. Attackers observe the mobile device screen while the legitimate user draws the secret graphical pattern for authentication. They thereby obtain knowledge of the pattern and the ability to perform replay attacks using this knowledge. Smudge attacks are a form of attacks specific to drawing based graphical passwords. Attackers obtain the mobile device after users have authenticated to it (it does not matter if the device is locked or unlocked then). Attackers then screen the device display to observe the residual smudge that might remain on the display (cf. [17, 309, 378]). This smudge might clearly indicate the graphical pattern used to unlock the mobile device (Fig. 3).



(a) Graphical pattern

(b) Residual smudge

Figure 3: Residual smudge on a mobile device display after performing a graphical pattern based unlock [378].

There exist approaches to modify graphical pattern authentication to become resistant to smudge attacks. One such approach is SmudgeSafe [309] in which a graphical pattern is drawn on top of a displayed image. For for each authentication a random rotation of the image is used. This leads to the smudge remaining on the mobile device display being uncorrelated to image rotation and thereby the pattern that needs to be drawn with the next authentication attempt. The authors find that SmudgeSafe significantly improves security over PIN or regular graphical pattern authentication in terms of smudge attack resistance. Drawbacks of this approach include reduced entropy in case of smudge attacks and increased user attention and duration to perform the authentication. With attackers taking the observed smudge into account the underlying entropy is reduced to the granularity of image rotation in combination with the allowed drawing accuracy of the graphical pattern. In combination with a graphical password image analysis as in [89] or [340] the most likely used rotation of the image for the observed smudge might easily be derived. Furthermore, the authors do not state the average duration their approach requires for device unlocks. It is reasonable to assume that the duration is higher than with classic graphical pattern authentication, as users at first need to recognize the image rotation and only then can draw their graphical pattern on the image. This also leads to higher required user attention as users now necessarily need to look at the device screen to observe the image rotation before they can draw their graphical pattern to perform authentication.

A similar approach has been investigated by Zezschwitz et al. [378]. They add randomization to graphical patterns to obtain smudge attacks resistance. They tried four slighty different approaches: marbles, compass, dial, and pattern rotation. With marbles the dots of the graphical pattern are arranged in a circular manner and users have to drag dots in the correct order towards the screen center. The resulting smudge always looks similar independently of the order of dots. With compass the circle instead is randomly rotated for each authentication attempt and users need to connect dots in the correct order. With dial the dots are represented by numbers and users need to perform a "dial" movement similar to dialing with old telephones. The password thereby is represented by the amount and order of numbers. The dial movement further leads to residues being wiped with each new movement which the authors refer to as consecutive blurring of residues. With pattern rotation a 3×3 dot pattern is presented with arbitrary rotation and users have to draw their pattern according to this rotation. Users have to determine the rotation of the dot arrangement by an additional compass symbol. This makes the approach comparable to SmudgeSafe [309]. The drawbacks with those approaches lie with authentication duration and error. As with SmudgeSafe, users need to dedicate more attention to authentication as they need to e.g. determine a rotation before performing authentication in a rotated manner. This leads to increased authentication duration and error [378].

Another approach is taken in [8] where wiping the screen is added as final step to perform authentication. While doing so provides for smudge attack resistance, its drawback is that users are required to perform an additional action for authentication that prolongs the total authentication duration and is difficult to perform with one hand alone. A different approach is taken in [85] where graphical pattern authentication is combined with an additional layer of implicit security using pattern dynamics. Thereby, additional factors like speed or pressure of drawing the pattern are considered to estimate if the pattern is really drawn by the legitimate user. The drawbacks of such approaches are twofold: a) users are required to train the mobile device to recognize their pattern dynamics. Such pattern dynamics can lead to legitimate users being rejected and further change over time (thereby requiring either retraining or online learning for continuous functionality). b) the approach does not prevent smudge attacks by design but adds a layer of security that cannot be attacked by smudge attacks. Consequently, when attackers successfully performed a smudge attack and obtained the graphical pattern the security of the approach is reduced to the security of the pattern dynamics recognition alone.

Further adaptations of graphical pattern authentication for mobile devices exist. One such approach would be [86] where the authentication is performed using special hardware with fingers on the backside of the device. The authentication secret thereby consists of a number of horizontal and vertical strokes performed on backside of device. The advantage of this approach is that it is more difficult to attack using shoulder surfing or smudge attacks than other graphical pattern based approaches. The drawback is that authentication takes significantly longer (about 4.5 s on average using complex, self-chosen secrets) and that special hardware capable of sensing finger movement on the backside of the device is required.

Summarizing, nearly all proposed knowledge based authentication approaches have some basic attributes in common. Their common major advantages are twofold. Most mobile knowledge based authentication approaches are purely software based. Thereby they can be implemented on most mobile devices with user interfaces without requiring special hardware (only certain approaches need mobile hardware with additional non-standard capabilities). Further, knowledge based secrets can easily be exchanged with a new secret in case the secret is disclosed and attackers could have obtained knowledge about it. In terms of drawbacks, knowledge based authentication approaches necessarily bear cognitive load on users. This is the case even when e.g. graphical secrets are used, such as with graphical passwords or patterns. Though the memorability of such secrets is increased compared to classic PINs or passwords users still have difficulties memorizing and recalling their secrets – especially with complex secrets and when using multiple devices. This is why users show tendencies to choose weak knowledge based secrets and reuse secrets across mobile device even with more easily memorable approaches like graphical passwords or patterns. These further lead to prolonged durations with many approaches requiring between 5 s and 20 s to authenticate. Additionally, mobile knowledge based authentication requires user attention in the form of users looking at the device screen while authenticating. From an attacker's perspective, many mobile knowledge based authentication approaches can be attacked with shoulder surfing or smudge attacks. In case the approaches are designed to be less vulnerable or resistant to these they usually add additional authentication effort, e.g. with requiring additional actions, prolonging the authentication duration, or increasing the false negative rate.

Hence, while knowledge based authentication is important not only for classic computers but also for mobile devices, there is a need for alternative forms of mobile authentication that are less obtrusive and bear less cognitive load on users. This is why biometrics and token-based authentication approaches that aim to be less in obtrusive in the mobile domain are discussed in the following sections.

3.2 BIOMETRICS BASED AUTHENTICATION

Biometrics are the second most important and widely used authentication approach with computer related systems. They use biometric properties of users (inherence) to perform authentication. These range from fingerprint or face to palmprint or vein authentication [101, 184, 276]. Biometrics have a number of notable advantages and disadvantages over knowledge based authentication. Most importantly, biometrics do not require users to choose or remember secrets for authentication. Thereby, users cannot choose weak secrets in the first place that would facilitate attackers using brute force or guessing. This also implies that no cognitive load is imposed on users independently of the amount of devices authentication is used on. Further, biometrics cannot be forgotten or lost, in contrast to knowledge or authentication tokens. However, biometric authentication may require user attention and time, and the consequences of biometrics used as authentication secrets being disclosed to third parties is more severe than with knowledge or token-based authentication. In this section we highlight aspects and mechanisms of biometrics important to mobile authentication. Certain aspects of face and gait biometrics are discussed in more detail as those biometrics are used exemplarily with the evaluation of our approach (Cha. 5).

Biometrics used for authentication can be categorized using different attributes [87, 167, 231, 276]. One distinctive category is biometrics either being physiological or behavioral. With physiological biometrics a physiological property is used for authentication. Examples include fingerprint or face authentication and do not necessarily require users to perform any action. In contrast, with behavioral biometrics *how* something is done is used for authentication, which usually requires users to perform an action (e.g. speaking, walking). Examples would consequently includes speaker or gait recognition. Another categorization is if biometric authentication is explicit or implicit. With the first explicit/active user interaction is required for authentication. An example would be users positioning their iris in front of an iris scanner. In contrast, with implicit authentication, authentication is performed without users explicitly or actively performing authentication. An example would be users being authenticated by their mobile devices while walking using gait biometrics. Implicit authentication has the advantage of requiring less user attention, therefore enabling less obtrusive authentication. One further categorization is if biometrics are strong or weak [167, 252]. Strong biometrics usually lead to a high confidence in the recognition or authentication result, while with weak biometrics confidence in results usually is lower. Examples for strong biometrics include fingerprint or iris and examples for weak biometrics include voice or gait. Other important properties of biometrics include continuity and obtrusiveness of biometrics [10, 88]. With continuous biometrics recording has to be done in a continuous manner. Examples would include sensor time series recordings for voice or gait biometrics, leading to samples possibly being of different lengths. In contrast, with non-continuous biometrics a sample is recorded non-continuously at a certain point in time, such as with taking a face or iris image. While there is a correlation of behavioral biometrics being continuous, physiological biometrics could be utilized in either continuous or non-continuous manner. An example would include face authentication which could be done either in a non-continuous – possibly explicit – form or in continuous – possibly implicit – form. The obtrusiveness indicates the effort users need to explicitly dedicate to authentication when using certain biometrics. While low obtrusiveness is desirable it is not always feasible. For example, while gait authentication could be done without users being required to explicitly dedicate any actions to authentication, iris authentication most likely requires users to position the eye with respect to the sensor position and to look into the sensor. Though some biometrics can distinctively be assigned to one of two of the above groups, many biometrics can be utilized in different ways and can therefore be assigned to multiple categories (e.g. speaker authentication could both be done explicit when requiring user to read a challenge out aloud or implicit while users are on the phone).

Biometric recognition and authentication most commonly used on mobile devices include fingerprint (e.g. Apple TouchID) and face (e.g Android Face Unlock). We now briefly review a number of biometrics that could be used with mobile devices, specifically pointing out the applicability with respect to mobile recognition and authentication.

3.2.1 *Fingerprint*

Fingerprint recognition is a strong, non-continuous, and mostly explicit physiological biometrics. It can be considered the most mature biometrics in both research and industrial applications with offline (print, scan) as well as online (live sensor) approaches [54, 69, 87, 214, 273, 376]. Fingerprints consist of different shapes and forms, such as whorls, plain and tented arches, or left, right, and twin loops (Fig. 4) [118, 142] which are either used directly for fingerprint recognition or on which feature derivation is applied first to subsequently perform feature matching.

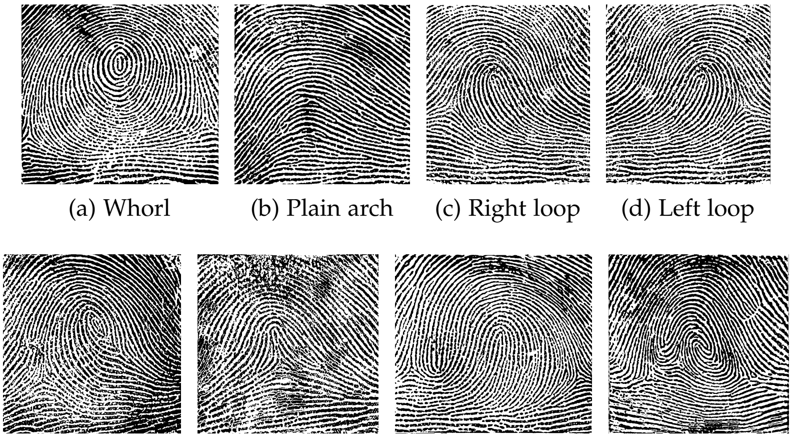

(a) Whorl (b) Plain arch (c) Right loop (d) Left loop
(e) Central pocket (f) Tented arch (g) Twin loop (h) Accidental

Figure 4: Henry's fingerprint classes [142] with different forms and shapes of fingerprints (adapted from [373]).

Different categories of features have been considered in the past based on singular points, orientation maps, global ridge structure, ridge frequencies, graphs, or syntactic approaches [6, 116, 373]. To increase image quality before performing feature extraction, different image enhancement approaches are frequently applied [137]. Recognition and classification of fingerprints and their features have been based on different models and matching approaches, including syntactic pattern recognition, graph matching, heuristics using singularities and/or (global) ridge structures, or classic pattern recognition models such as support vector machines (SVM), neural networks (NN), k-nearest-neighbor (KNN) models, or hidden Markov models (HMM) [116, 188, 373]. Like other biometrics, fingerprints can be spoofed [266]. Materials used for fingerprint spoofing range from gelatin over Play-Doh to silicone and have been shown to successfully trick even commercial fingerprint authentication systems using cheap materials [217]. Anti-spoofing for fingerprint ranges from hardware based to more widely applied software based approaches [69, 217]. Sophisticated attackers with knowledge about the used system and access to high cost spoofing materials seem to be able to also trick anti-spoofing fingerprint recognition approaches, which indicates that this is still an unsolved problem [320].

With mobile devices either touch-based embedded sensors or embedded cameras can be used for fingerprint authentication. Touch-based fingerprint sensors have the advantage of enabling fast authentication (around 1 s for capturing the fingerprint [123]). They further are less obtrusive as the fingerprint sensor can e.g. also function as the device button to turn on the screen – which limits the additional effort for users making sure that the finger is well pressed to the sensor [123]. The drawback of this type of sensor comes as additional costs as most mobile devices do not yet ship with embedded fingerprint sensors. This could be explained by the hardware being fingerprint specific, thereby not being reusable for other tasks, like embedded cameras. In contrast to touch-based sensors, embedded cameras are shipped with most mobile devices already, therefore no additional hardware is required. The drawback of fingerprint authentication using embedded cameras comes with increased user effort. Users need to position their fingerprint in front on the camera, ensure a sharp image (e.g. no motion blur), and a correctly illuminated fingerprint for authentication to work. One such approach would be the touch-less fingerprint system in [282]. They use cameras (e.g. of mobile devices) to capture fingerprints and perform authentication without users being required to touch any fingerprint sensor. In their evaluation some illumination conditions turn out to be difficult to perform authentication. Though they do not consider authentication duration the total required duration can be assumed to be higher than with touch-based sensors.

3.2.2 *Face*

Face recognition is a strong, both explicit and implicit, non-continuous and physiological biometrics that recognizes individuals by their faces (Fig. 5).

Both geometry and appearance based approaches have been used to perform 2D face recognition and authentication [185]. Geometric approaches derive facial features and key positions in face images, then decide on recognition or authentication using this information. In contrast, appearance based approaches derive features directly from the pixel representation of face images without considering facial features directly. In the past, a considerable amount of appearance based face recognition and authentication approaches

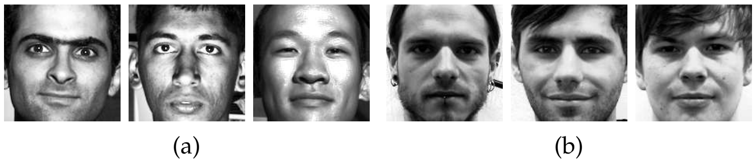


Figure 5: Face image samples from (a) the Yale-B face database and (b) the Panshot Face Unlock database.

has been discussed (cf. [1, 41, 380, 382]). Important examples include Eigenfaces [314, 357, 369], based on which further simple yet effective dimensionality transformation and reduction approaches have been proposed for face recognition and authentication, such as linear discriminant analysis (LDA) [204] or Fisherfaces [26, 369]. Further approaches additionally employ other models, such NNs [97] or SVMs [313], or different appearance based feature derivation procedures, such as local binary pattern [346, 369] or wavelet transformation and related approaches [201].

Mobile face recognition has been demonstrated to be feasible using different features and models, e.g. yielding about 10-11% HTER [219] on the MOBIO database [218]. Besides the more widely employed explicit face authentication also implicit continuous mobile face authentication approaches has been investigated [73, 298]. One notable advantage of mobile face authentication thereby is that is can be performed without requiring additional or uncommon hardware in mobile devices, as most devices feature cameras of sufficient quality.

3.2.3 *Iris*

Iris recognition is a strong, non-continuous, and mostly explicit physiological biometrics that distinguishes people using the unique patterns of the human iris [42, 79, 223]. In general, iris recognition processing chains including iris detection, segmentation, preprocessing, feature derivation, and matching of iris images [80]. The main advantage of iris recognition over other biometrics is the distinctiveness of the underlying biometrics [81, 82]. For example, Daugman [81] estimates the false positive rate to be in between $\frac{1}{5\cdot 10^{15}}$ and $\frac{1}{10^{6}}$ depending on the configuration of the authentication approach, based on 200 billion iris comparisons. To obtain good results, most iris recognition approaches rely on using near infrared (NIR, around 700-900 nm) illumination and cameras [80]. The reason is that human eyes contain melanin that blocks visible light. Depending on the amount of melanin the iris pattern might partially or completely remain hidden in visible light. However, melanin is transparent in the NIR spectrum (Fig. 6), which is why iris recognition frequently relies on NIR light sources (the NIR part of visible light is usually too weak to allow for good iris recognition results) and NIR cameras.

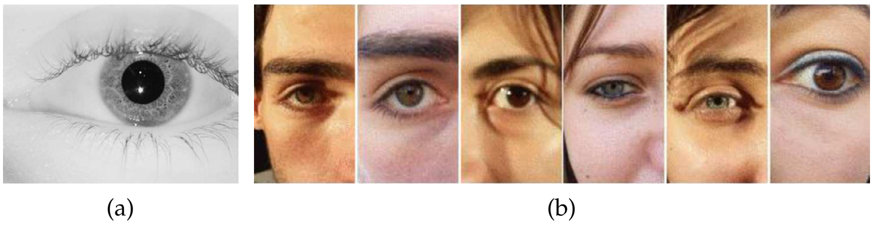


Figure 6: Iris samples recorded using (a) a NIR light source and high quality NIR camera (adapted from [82]) and (b) visible light and mobile device cameras (adapted from [221]).

While the iris can be considered one of the strongest biometrics available in the mobile environment today, its drawbacks come from a potential for obtrusiveness and additional costs for NIR hardware. With most iris recognition approaches on mobile devices, users have to look straight into the camera. This requires additional user attention and time (mobile iris authentication was measured to be around 1.8 s to 4.2 s in [261]). Further, similar to touch-based fingerprint sensors, most mobile devices are not equipped with NIR light sources and NIR cameras. Embedding this hardware in mobile devices is associated with additional costs. Though there exist approaches using cameras that work in the visible light spectrum and that are shipped with off-the-shelf mobile devices subsequent iris recognition remains difficult. Examples include [268] which use a white LED for iris illumination, or [22], which find their approach to work on data from iris reference databases such as UBRIS [265] or UPOL [90] but to yield non-optimal results when applied to uncontrolled mobile iris samples in the visible light spectrum from the MICHE-I database [221]. Another complicating factor with mobile iris recognition in the visible light spectrum are reflections, e.g. on the eyeball or glasses users are wearing, which need to be addressed accordingly [261].

3.2.4 *Gait*

Gait is the way humans walk [244, 352] and can be used for recognizing and distinguishing individuals [364]. Gait biometrics are most frequently considered a weak, continuous, implicit, and behavioral biometrics. Gait recognition and authentication [190] can be based on different types of data, including visually [299] or floor sensed information [233] (e.g. humans recorded in context of CCTV surveillance or sensors being embedded with floors humans walk on, such as pressure sensors) as well as information from sensors worn by humans themselves [113]. With the latter, different sensor types and sensor positions on the human body have been utilized [115]. Mobile devices like smartphones have become a powerful source of such data as they feature a number of different sensors and are frequently with people while they are walking (e.g. inside a trousers pocket). Especially accelerometers shipped with mobile phones have been used for acceleration based gait authentication [331]. As human walk is of cyclic nature, each step can be seen as repetitive cycle (Fig. 7).

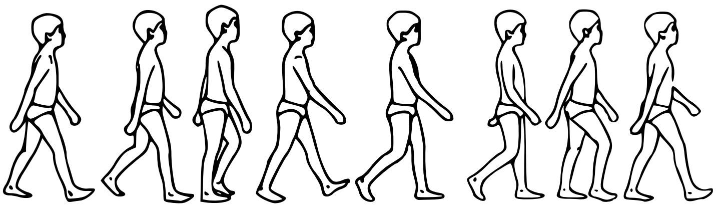


Figure 7: Visualization of the cyclic nature of human gait [352].

With acceleration based gait authentication both cycle and window based approaches have been utilized in literature [144]. With cycle based approaches individual step cycles are segmented from recordings and used for subsequent recognition. Analogously, with window based approaches, a (possibly fixed length) sliding window is used on recordings instead to segment data chunks.

The matching procedure of acceleration based gait authentication often involves dynamic time warping (DTW) as distance metric between two time series [215, 241, 358]. Regular DTW thereby brings a memory complexity of at minimum $m \cdot n$ for two time series of length m and n. For acceleration based gait authentication without using DTW, various features have been used. Those include: average, median, min, max, standard deviation (SD), and median absolute deviation (MAD) acceleration of individual axes and their magnitude [190, 254], root mean square (RMS) acceleration [254], mean- and zero-crossings [254], principal component coefficients of acceleration [39, 321], binned acceleration distribution [113, 190, 254], time between peaks [190], discrete cosine and fast Fourier transformation coefficients [18, 114, 159, 291], and Mel- and Bark-frequency cepstral coefficients [144, 254]. Further, wavelet transformations have been used with non-cycle-based acceleration gait data [144, 267] and floor sensor based gait data [238], as well as on acceleration based gait style recognition [163], which in contrast to gait identification or authentication does not distinguish individuals but gait styles. On those features, again a number of non-DTW based models have been applied, including cross-correlation based [216] or tree based models [190], NNs [190, 303], SVMs [254, 321], analysis of variance (ANOVA) [18], Gaussian mixture models (GMM) [159], and HMMs [254]. Finally, one advantage worth mentioning is that acceleration based gait authentication can be performed with off-the-shell mobile devices containing acceleration sensors, without requiring additional or uncommon hardware [243].

3.2.5 *Speaker*

Speaker recognition is a continuous behavioral biometrics used both explicitly and implicitly that recognizes humans by their voice and has been well researched in past decades [24, 141, 192, 284] (Fig. 8).

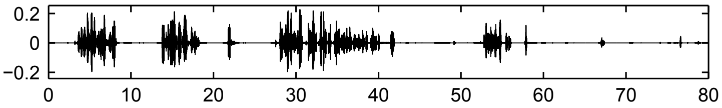


Figure 8: Speech waveform as microphone recorded excess pressure over time [179].

Speaker recognition can be grouped into being text dependent [191] or text independent [179]. With text independent approaches users are recognized independently of which words or text they speak. On the one hand, users could conceptually be continuously authenticated using voice in everyday situations without using predefined phrases or being required to read text before authentication. On the other hand, attackers could possibly easily record a user's voice and perform a simple replay attack to trick the authentication. With text dependent approaches users have to speak a predefined text. This could e.g. be a phrase known in advance or a text displayed on a screen. The advantage is that the recognition system knows the text to be spoken and therefore can compare spoken words with text. This could be used as an advantage to e.g. incorporate knowledge based secrets with speaker recognition (requiring users to speak a shared secret phrase), or to require users to read different phrases from a screen for different speaker authentication attempts. The latter would be a means of preventing the simple replay attacks mentioned above [19]. Notable technical details of speaker recognition include Mel-frequency cepstrum coefficients (MFCC) as features to characteristically represent speakers [138], and GMMs e.g. with universal background models (UBM) [52]. The latter are used to first learn both the target speaker and independent speakers/background voices and noise, then to compute the likelihoods of the present audio being the legitimate user speaking or it being background voices/noise. The resulting probability ratio can then be used to yield a recognition or authentication decision. Challenges to speaker recognition arise with little available data for training models, noisy environments, an increased distance between speaker and recording microphone, as well as unfavorable positioning of the microphone or angle towards the speaker [99, 234, 251, 271].

With mobile devices, speaker recognition could be used to continuously authenticate users. This requires few to no user attention (e.g. while being on the phone) and specifically does not require users to look at the mobile device screen for authentication. The latter would also be true with explicit speaker authentication. However, using challenge based explicit speaker authentication to prevent replay attacks [19] would increase the corresponding user effort as users also have to read phrases or know them in advance for authentication. In terms of applicability mobile speaker recognition has been shown to be feasible for different approaches (including GMM-UBM and SVM models), e.g. yielding around 10-11% HTER [219] on the MOBIO database [218]. It further has been shown that continuous speaker recognition with low energy requirements is possible using low power (co)processors in mobile devices [203].

3.2.6 *Signature*

A less frequently discussed example of mobile biometrics is signature recognition. Signature recognition is a special case of handwriting recognition which tries to recognize individuals based on their handwriting. In contrast, signature recognition only considers signatures of users but is based on the same concepts, often using the same underlying mechanisms for processing and classifying data. Signature recognition could thereby be categorized as explicit and non-continuous behavioral biometrics. For mobile devices, signature recognition could be interesting as additional biometrics modality for explicit authentication, e.g. remotely signing contracts with a signature. For handwriting and signatures both offline and online recognition have been explored [264]. Offline recognition is based only on the final handwriting or signature (e.g. using a scan). With online recognition, additional features are available such as timing, speed, or pressure applied to a digital screen (Fig. 9). Using this information, different preprocessing, feature derivation, feature selection, and classification approaches have been explored [59, 308, 343].

On mobile devices, DTW has frequently been employed as signature matching algorithm [33, 157, 187, 344]. Mobile device signature capturing capabilities seem limited over dedicated stationary hardware, which negatively influences signature recognition accuracies [157]. Using smaller devices (e.g. phone sized) seems to yield better signature recognition results than using larger devices (e.g. tablet sized) [33]. Further, using a pen or stylus yields better signature recognition accuracies than using a finger on a capacitive display [33, 344] though finger drawn signatures seem harder to attack than pen-based signatures with zero-effort attacks [344]. If signatures are used across different devices or different modalities (finger, pen) recognition results become worse [33]. One notable advantage of signature

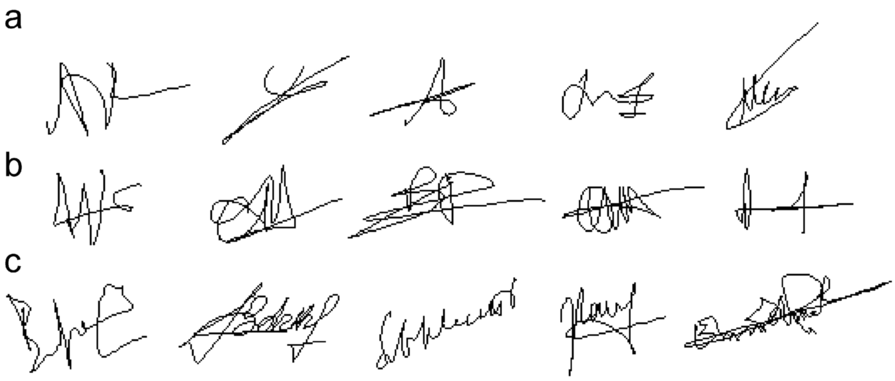


Figure 9: Signatures from the DS2 signature database with personal entropy ranging from high (a) to low (c) [157].

recognition is that it can be performed on off-the-shelf mobile devices without requiring additional or uncommon hardware, similar to face and acceleration based gait authentication.

Summarizing, biometrics can be assumed to be less obtrusive than knowledge based authentication as they do not bear cognitive load on users and cannot be forgotten or lost. However, biometrics need thorough protection from disclosure as they cannot easily be changed after being disclosed. This protection challenges the design of biometrics matching approaches and might result in decreased performances (Sec. 3.3). Many biometrics are applicable in some but not all authentication situations. For example, face or iris authentication might not be applicable in all illumination conditions, while voice authentication might not be applicable in noisy environments. However, the combination of such biometrics can result in robust authentication where users could choose biometrics best suitable in different authentication situations. Therefore, each biometrics usable on mobile devices represents one option to perform authentication. The more such options are available, the less obtrusive the overall authentication process is. This especially includes biometrics that cause virtually no authentication overhead but are only applicable in few situations (e.g. gait authentication while walking). Besides biometrics such combinations can also incorporate knowledge or token-based authentication to provide further options and consequently further reduce the overall obtrusiveness.

3.3 PROTECTING BIOMETRICS IN MOBILE ENVIRONMENTS

Biometric authentication uses physiological or behavioral characteristics for authentication. Thereby these characteristics become the authentication secret that should be protected adequately, similar to secrets with knowledge or token-based authentication [36, 165–167, 252]. To authenticate, biometrics of the legitimate user have to be recorded and stored with the authentication system (or authentication service) at first. For authentication, users record new biometrics samples which are matched with the stored templates to derive an authentication decision. The biometric information could thereby be disclosed to unauthorized third parties on multiple ways [166, 275]. Amongst others these include a) devices storing templates being lost or stolen. b) devices processing or storing templates being infected with malware that transmits information to attackers. c) templates being derived from public information or deliberately recorded by attackers. d) authentication services (e.g. centralized databases) being accessed by third parties, thereby biometric templates stored there potentially falling into the hands of attackers. On mobile devices, biometric templates could be extracted from the recording at sensors to the storage and matching procedure [275], depending on the capabilities and access rights of attackers. In contrast to knowledge or token-based secrets, biometric templates have more severe consequences if they are disclosed. While knowledge and token-based authentication secrets could easily be changed (e.g. remembering a new secret or acquiring a new token), biometrics cannot easily be changed. Consequently, biometrics disclosed once might need to be considered compromised forever. This makes biometric templates a potentially high value target for attackers and bears additional risks for user relying on biometric authentication [165, 166]. However, using obtained biometric templates for authentication is more difficult for attackers than using e.g. an obtained knowledge based secret. After obtaining biometric templates, a reconstruction of the biometrics has to be created which can be presented to the sensor for authentication. This reconstruction increases the effort of attacks, thereby makes attacks more difficult. Still, attacks using reconstruction of biometrics from templates have been demonstrated for different biometrics. Examples include the reconstruction of fingerprints from stored minutiaes [53, 55, 292], irises from iris codes [353], or faces from Eigenvalues using hill climbing attacks [5]. Mobile devices being more easily lost, stolen, or accessed by third parties while unattended by the owner than classic computers further emphasizes the need to protect biometrics used for authentication on mobile devices.

3.3.1 *Biometrics and Classic Cryptography*

Neither classic cryptographic en- and decryption nor classic cryptographic hashes are sufficient to protect biometrics due to the avalanche effect. With the avalanche effect, a bit flip in clear text data ideally leads to 50% flipped bits in the resulting ciphertext or hash [341]. Biometric templates of the same user differ slightly for different readings by design. The ciphertexts and hashes of such slightly different samples are completely different due to the avalanche effect. Therefore, even with multiple biometric templates of the same user being similar to each other, their ciphertexts and hashes are intentionally unrelated to each other. This makes comparison or matching of ciphertexts and hashes of biometric templates for authentication infeasible. When using encryption, templates could be decrypted to be matched with new samples. This needs to be done each time an authentication attempt is made. Therefore, if attackers obtain control over a mobile device they could access unencrypted templates each time an authentication attempt is made. Further, the decryption secret could fall into the hands of attackers which would enable them to also decrypt templates obtained long ago. Using en- and decryption of biometric templates therefore does not solve the problem of protecting biometrics but only changes the goal to protecting the decryption secret instead. This is why biometrics can be protected using either the algorithmic approach of biometrics template protection or secure hardware such as smart cards (SC).

3.3.2 *Protecting Biometrics with Biometric Template Protection*

Biometric template protection is an algorithmic approach (not using secure hardware) towards protecting biometrics from disclosure. Biometric template protection can be categorized in biometric cryptosystems and cancelable biometrics. Both approaches have in common that they never store biometric templates or features derived from them in their original form to avoid them being disclosed. We give a brief overview of both categories in this section. For a more comprehensive review we refer to the surveys and books on biometric template protection, including Breebart et al. [43], Cavoukian and Stoianov [57], Jain et al. [165, 166], Ngo et al. [252], Patel et al. [263], Rathgeb et al. [280, 281], and Uludag et al. [348, 349].

3.3.2.1 *Biometric Cryptosystems*

Biometric cryptosystems combine biometrics with cryptographic keys. They can be categorized in either key-binding or key-deriving biometric cryptosystems [57, 166, 281]. Key-binding biometric cryptosystems at first bind a cryptographic key K with biometric samples S_a to create helper data H. New biometric samples S_b that are sufficiently close to S_a can in combination with H be used to release K. In contrast, key-generating biometric cryptosystems derive helper data H directly from S_a, from which a cryptographic key K can further be derived. H can be discarded after deriving K or it can be stored to assist when deriving K from S_b. Both forms of biometric cryptosystems have in common that H is (potentially) stored – instead of storing biometric templates themselves. In order to protect users' biometrics, as a consequence H must not enable attackers to derive the original biometrics in case it is disclosed. One drawback with biometric cryptosystems comes from the limitation of entropy of K from the underlying distribution of biometric samples. It has been shown that for K to be random, its maximum length L_K is bounded by the false positive rate (FPR) of the biometrics resulting from zero-effort attacks, which needs to be seen as significant drawback for authentication or cryptographic purposes (Eq. 4) [47, 49, 281].

$$L_K \leqslant -\log_2(FPR) \tag{4}$$

Well known examples of key-binding biometric cryptosystems include the Fuzzy Commitment Scheme [174] and Fuzzy Vault [173]. With the first, error correcting functionality is used to enable any S_b sufficiently close to S_a to release K. The latter extends this concept by using polynomial reconstruction for releasing K from H and S_b. Both approaches have been applied to different biometrics in the past (cf.[154, 160, 175, 208, 232, 246, 249, 250, 278] and [110, 176, 197, 232, 247, 330, 371, 374]). Well known examples for key-generating biometric cryptosystems include fuzzy extractors (e.g. [47, 92, 93]) and secure sketches (e.g. [327, 328]). With the help of H both extract information from S_a and S_b instead of binding K into H. Fuzzy extractors thereby reliably extract K from both S_a and S_b. H assists in the reconstruction/correction of errors in K resulting from the difference between S_a and S_b. In contrast, with secure sketches the exact original sample S_a is used as basis for K (e.g. using cryptographic hashing). H thereby assists in the reconstruction of S_a from S_b. Again, both fuzzy extractors and secure sketches have been applied to different biometrics in the past (cf. [12, 14, 46, 48, 327, 328, 345, 372]).

3.3.2.2 *Cancelable Biometric Templates*

In contrast to biometrics cryptosystems, cancelable biometrics do not bind or derive a cryptographic key from biometric samples. Instead, they transform biometric samples before storing them so that they can still be compared/matched in the transformed domain, but also so that it is hard for attackers to derive the original samples from the transformed templates [166, 167, 263, 281]. Cancelable biometrics can be categorized in either non-invertible transformations or biometric salting. With the first, biometric templates are transformed using a non-invertible transformation to obtain secure templates. In contrast, biometric salting utilizes transformations that are conceptually invertible (this focuses on the transformation itself and does not imply that deriving the original biometric template from the secure template is necessarily feasible [281]). On the one hand, in order to protect biometrics with biometric salting the used transformation parameters must be kept secret, which needs to be considered as significant drawback over non-invertible transformations. On the other hand, non-invertible transformations show noticeable performance degradations (both decreased accuracies and increased computational requirements) over biometric salting and regular biometric systems [281].

Important examples of non-invertible transformations include the original proposal by Ratha et al. [272] which use image-based block permutations and surface-folding in order to obtain revocable biometric templates. Further examples include the application on signature biometrics by Maiorana et al. [209–211] and the alignment free approach on iris biometrics by Rathgeb et al. [277, 279]. Non-invertible transformations have further been applied to different biometrics (cf. [193, 212, 383]). Biometric salting too has been applied to different biometrics (cf. [235, 259, 335, 359, 383]). Notable examples include BioHashing [121] which can be used in two factor authentication manner.

3.3.3 *Protecting Biometrics using Secure Hardware*

The second option to protect biometrics on mobile devices besides biometrics template protection is using secure hardware to process and store biometric information. This option seems to have received less attention in literature in the past and is the option used with mobile biometric authentication in our approach (Cha. 5).

3.3.3.1 *Smart Cards*

Smart cards (SC) [269] are special integrated circuits which provide certain characteristics that are useful for security sensitive applications: a) cryptographic operations (e.g. encryption, decryption, hashing) can be performed directly on the chip, often in hardware. b) SCs are intentionally kept small and less complex to make unintended behavior/bugs in the system less likely. That is, it is easier to verify that there are no major security flaws. c) data and application code in the memory is protected against unauthorized access and tampering. A serial interface, which is controlled by the operating system of the hardware, is the only way to access this data.

However, besides those advantageous characteristics, SCs also bring limitations that need to be considered for applications relying on them: a) data transfer to/from SCs being restricted in bandwidth (cf. Hölzl et al. [156] with measurements of 329 B/s for contactless and 3,31 kB/s for contact cards). b) while some modern SCs already use a 32 bit architecture, many currently deployed cards are still based on a 16 bit architectures. That is, there are no 4 byte integers and integer calculations in hardware on those cards. c) persistent and volatile memory are highly limited with a maximum capacity of around 1 MB for current cards. d) finally, SCs are limited in computation capabilities: for example, there are no native floating point operations available in hardware. Computations performed in software are considerably slower than on PCs or mobile devices due to clock rate of SCs usually being in the MHz range.

With biometric authentication these computation and data transfer limitations affect both the internal structure of authentication models and number and type of features that can be used with SCs. For example, using 4 byte integers in a 16 bit environment requires more complex data structures in internal computations (i.e. operations on arrays for simple multiplications). Hence, using small value ranges for both model representation and features transferred to the SC are preferred. Further, transmission bandwidth to/from the SC is limited, which limits the amount of data that can reasonably be sent to the SC during user authentication.

3.3.3.2 *Biometrics with Mobile Devices and Smart Cards*

Smart cards (SC) are frequently shipped in off-the-shelf mobile devices in the form of secure elements (SEs). These can either be directly embedded in the phone hardware, extended with an SD card, or provided within modern SIM cards [156]. With biometrics on SCs, the storage and matching part can either be achieved with template-on-card (TOC) or match-on-card (MOC) techniques (cf. [32, 44, 75, 166, 167]). With TOC, biometric templates of the user are recorded by sensors of the mobile device and stored on the smart card during enrollment. During authentication the enrolled templates are fetched from the SC and compared with new recordings outside the SC. In contrast, with MOC authentication, new recordings are transfered to the SC and compared with previously stored templates directly on the SC.

This leads to the following noticeable differences of MOC over TOC: on the one hand, after a user's biometric templates have been stored on the SC during enrollment, they never leave the SC. Hence, MOC reduces the possibilities for leakage or theft of biometric templates over TOC. On the other hand, comparing users' biometric templates with new biometric recordings on the SC is subject to hardware limitations of the SC, namely transfer bandwidth to and computational limitations on the SC. Hence, the portion of data that can be transfered to the SC and the computations that can be done on the SC have to be selected carefully. As reducing the risk of leakage or theft of biometric templates is important, MOC is regularly preferred over TOC, despite the accompanying computational limitations. In turn, these limitations lead to restrictions in how existing MOC approaches are frequently designed (cf. [32, 66, 125, 166, 167, 260]):

- MOC approaches usually rely on restricted operations and logic for matching templates with new recordings. Hence, they often do not utilize regular, offline trained machine learning (ML) models. Further, they are frequently restricted to a small set of – sometimes handpicked – features to be used in the matching process. Both necessarily limit the MOC discriminative power.

- To reduce computational requirements, most MOC operations are very domain specific. The underlying mechanisms are usually strongly adapted to the used biometrics. This impedes the adaption of new biometrics in MOC approaches, where it would be beneficial to have reusable concepts for feature derivation, model representation, and matching operations.

3.3.3.3 *Previous Work Using Match-on-Card Authentication*

To this date, fingerprints are the best researched biometrics with MOC authentication approaches. They usually utilize small templates and a small amount of features (mostly minutiae based), which in turn lead to relatively simple matching procedures (cf. [32, 120, 129, 260]). MOC authentication with biometrics other than fingerprints has been covered by little research. Examples include Choi et al. [66], who use SVMs with a limited amount of features and FPGAs for speaker verification in a MOC manner. Czajka et al. [75] perform iris recognition by deriving a 1024 bit iris code from samples outside the SC, then match new recordings with enrolled templates on the card using a computationally lightweight Hamming distance. This approach is therefore more similar to fingerprint than e.g. face authentication in terms of template size. Another authentication related example is human identification from CCTV records [236]. Although the approach is conceptually similar to gait authentication from visual data (including the matching based on simple distance metrics), the processing chain, including used features such as cloth color and human height, represent a major difference. To the best of our knowledge there exist no approaches to acceleration based gait MOC authentication yet. With the majority of the described approaches (Sec. 3.2.4), either retraining the model for individual users would be required, or neither training the model, nor using a ready trained model to predict new samples is feasible on SCs with respect to their computation requirements. Still, similar feature derivation mechanisms can be utilized in MOC approaches as long as they are computed outside the SC.

Similarly, the computation of most described face recognition and authentication approaches (Sec. 3.2.2) would be infeasible with SCs and MOC approaches. Research towards face authentication with SCs mostly relies on using limited matching on the SC. For example, Tistarelli et al. [342] propose a face authentication TOC approach in which they use morphological filtering and adaptive template matching to extract the position of relevant facial features for matching. During matching they fetch enrolled templates from the card and compare them to new recordings using a space-variant approach based on principal component analysis (PCA). Lee and Bun [195] combine PCA projection weights, average intensity and edge values as features with genetic algorithms (GA) for feature selection. They thereby largely reduce the amount of features, which enables the usage of an SVM model for authentication. Kittler et al. [181] state that PCA compresses templates in a suboptimal way for usage on SC. They therefore propose a MOC approach using a 1D, client specific LDA, of which they utilize the distance of new recordings to both the stored client template and to the average impostor to derive a scalar distance measure. As tradeoff between computational requirements and authentication performance, Bourlai et al. [38] utilize the client specific LDA proposed in [181] as feature derivation mechanism, then use the vector dot product of a new recording and the enrolled template with a predefined threshold to obtain an authentication decision.

Summarizing, mobile biometrics could be protected using either algorithmic biometric template protection or secure hardware to store and process biometrics. The main advantage of biometric template protection is that no specialized hardware is required to be embedded or shipped with mobile devices. In terms of drawbacks, biometric template protection has shown performance degradations (in both decreased matching accuracies and increased computational requirements) over regular biometrics systems [281]. Further, both authentication as well as the protection of biometrics rely on the properties of the underlying algorithmic approach. In combination with the natural distribution of biometric samples this can lead to reduced entropy [47, 49, 183, 281], which could make attacks on the authentication system or deriving the original biometrics from protected samples easier for attackers. In contrast, secure hardware like SCs relies on the hardware being secure and difficult to tamper with to protect biometric templates. In terms of advantages, using SCs does not rely on algorithmic properties to protect biometrics. On the one hand it thereby does not imply computational overhead or degradation in matching accuracies caused by such properties. On the other hand the limited processing and storage capabilities of SCs are a challenge to designing suitable and well performing biometric authentication procedures. Further, SCs need to be embedded/shipped with mobile devices in order to be able to protect mobile biometrics. As some modern mobile devices as well as modern SIM cards start containing SCs this can be assumed to only be a small drawback.

3.4 TOKEN-BASED AUTHENTICATION

Token-based authentication is the third major way of authentication besides using knowledge and biometrics. With tokens, authentication is performed using “something users possess”. Thereby the token is a physical object in possession of legitimate users that either performs authentication or aids users in doing so, like a key to a physical lock.

3.4.1 *Functional Principle*

Authentication tokens come in different forms, mostly holding a secure storage containing an authentication secret. Well known examples include ATM cards, where the embedded chip contains the authentication secret, or devices for generating one time passwords (OTP), e.g. using a token device like YubiKey[2]. The latter thereby usually utilize one of two approaches for OTP generation: either being time synchronous, changing secret synchronous with a master [361], or using a challenge-response approach [256, 315].

From an authentication perspective, tokens provide a number of advantages and drawbacks over knowledge and biometrics based authentication. Similar to biometrics, tokens do not bear cognitive load on users for remembering the authentication secret. This is because the secret is held by the token itself. Instead, the cognitive load imposed on users is to bring the token for authentication. If the token is forgotten or not available, authentication becomes impossible. As tokens are better suited than human memory to store complex secrets and because tokens with embedded cryptographic hardware are better at performing cryptographic operations, token authentication typically features better security in terms of cryptographic strength. Consequently, the computational security with token-based authentication is usually quantified as cryptographic entropy (e.g. using a 256 bit key with an AES cipher [315, 341]), in comparison to limited entropy when using user chosen passwords [256]. As a result, token-based authentication is usually harder to guess using brute force attacks. Further, tokens often bear some physical attack resistance using special hardware that is difficult to tamper with and/or disables itself if tampering is detected [36, 256].

In terms of drawbacks, if a token is lost or stolen by attackers it is more difficult to replace than a knowledge based secret, but still easier than exchanging disclosed biometrics. Further, the acquisition of token hardware (including both the token device and the device reading the token) is usually associated with costs. Exceptions include when for example users already own all necessary hardware (e.g. the functionality required for reading the token being embedded with standard computers and the token device being e.g. an already owned smart watch). This also leads to additional costs each time a token needs to be renewed, e.g. after loss. Different token-based authentication systems further complicate this matter. There exist many different commercial token-based authentication systems. This might

[2] Yubico online presence for YubiKeys: https://www.yubico.com/products/yubikey-hardware/.

require users using tokens for different authentications to buy and carry multiple tokens, leading to increased physical and financial effort [256]. In terms of obtrusiveness with daily usage of authentication tokens, drawbacks are twofold: a) tokens need to be taken along which can cause additional effort and bear cognitive load on users to not forget them. This drawback can be relaxed by combining the token with something that is taken along anyway (for example keyrings, watches, or rings). b) performing token-based authentication usually takes some time. For example, when users want to use a YubiKey token to perform authentication (that is not stationary connected to a computer) they at first need to locate/grab the token and connect it to the device before they can authenticate.

From an attacker's perspective, in contrast to knowledge or biometrics based authentication, tokens can be physically lost or stolen. This is especially important for mobile devices: if both the device and the corresponding authentication token are with the user both could be obtained by attackers at the same time (e.g. theft of the mobile phone and the keyring holding the authentication token). This would allow attackers to authenticate to the mobile device. Because of tokens being portable, many token-based authentication approaches add an additional layer of security to the token itself. Such could be done by requiring the user to authenticate to the token using a knowledge or biometrics based authentication [256]. This makes the subsequent authentication using the unlocked token a two-factor authentication with all corresponding benefits in security and drawbacks in usability [315].

3.4.2 *Previous Work Using Token-Based Authentication*

With a focus on mobile environments there exist two major ways of token-based authentication: using a wearable or mobile device based token to authenticate to classic computers, stationary terminals, or similar (from now on referred to as "computers" only) and using a token to authenticate to mobile devices themselves, with the first having received more research attention in the past than the latter.

Most approaches using mobile or wearable tokens to authenticate to computers are based on wireless communication between token and computer using e.g. near field communication (NFC), radio frequency identification (RFID), IEEE 802.11 (WiFi), or Bluetooth (BT) [71, 133, 177, 257, 322, 336]. Thereby, different ways of protecting computers with tokens have been explored. In [71] a wearable authentication token is used to communicate with computers over short range wireless communication. They perform file en- and decryption on the computer using a secret from the token (files are encrypted when user leaves and decryption when user returns). Two tokens are combined for authentication in [177]. One token is used to unlock the second token based on proximity (e.g. wireless signals). The second token is only operable when unlocked with the first token and is responsible for performing the authentication itself. A wristband is used as authentication token in [257]. The wristband needs to be unlocked using fingerprint authentication before being usable. The wristband further checks for vital signs of its wearer to increase theft resistance (it could e.g. lock when detached). With PICO [322] a token with a "main" and a "pairing" button, a display, a camera, and a NFC interface is used. The token communicates to computers using NFC for both pairing (by pressing the "pairing" button) and authentication (by pressing the "main" button). In related approaches off-the-shelf mobile devices are employed as authentication tokens using a wireless connections to communicate with computers, e.g. by using BT [133, 336]. The main advantage of using regular mobile devices as token is that users do not need an additional device for authentication that they need to bring along or that could be lost or stolen.

The mentioned approaches – using or not using off-the-shelf mobile devices – bear a common drawback. Their security does not only rely on cryptographic communication security for communicating information between a paired token and the corresponding computer, or a physically secure token that does not fall into hands of attackers. Their security is also proximity based, that is authentication is only supposed to work when token and computer are within a certain range of each other. Consequently, authentication is possible within the corresponding range of NFC, RFID, or WiFi communication. Some previous research points out that the range of the used short range wireless communication is limited by design (e.g. from a few centimeters to a few meters) [253, 322]. Though this can be considered an advantage over not requiring authentication at all, attackers could amplify or forward received signals, or use bigger antennas and transceivers to extend the distance in which authentication work, resulting in man-in-the-middle attacks [196, 290]. Attackers could also obtain unauthorized access to devices as long as the token is within authentication range. An example would be attackers accessing a computer with the legitimate user having turned their back to them and the computer being unlocked by the token carried by the user. Such attacks have been demonstrated e.g. by Lee et al. [196] who conclude that authentication using wireless signals needs to be considered vulnerable to these attacks. They propose to use ultrasound instead of NFC, RFID, or WiFi to communicate between token and computer, as ultrasound is more difficult to relay or cancel by attackers.

Another approach that by design does not rely on wireless communication distance is using vibrations to communicate information. This has been demonstrated in two-factor authentication with a mobile device and an RFID token, where the mobile device is used to unlock the token that can then be used for further purposes [302]. Unlocking the RFID token thereby is done by pressing the token against the mobile device, thereby sending and receiving information over device vibrations. Another example would be to require users to read an authentication secret (possibly OTP) from the token and to enter it on the computer. This has been demonstrated e.g. by an online authentication service sending an authentication code to the mobile device using SMS over the GSM network, which the user enters on the computer to perform authentication [332].

In contrast to classic computers, mobile devices have rarely been addressed as the device to be authenticated to using tokens. Approaches doing this mostly rely on the same underlying mechanisms as the previously mentioned approaches. For example, wearable tokens to authenticate to mobile devices are used in [60, 74, 108, 119, 128, 253, 325]. Some approaches allow authentication when wireless communication is possible between token and mobile device based on NFC [60, 108] or combine NFC with the requirement of matching locations (e.g. via GPS receivers, which requires both mobile device and token to have sensors to independently determine their location) [119]. Other approaches allow authentication to mobile devices when the token is within BT communication range [186] or rely on proximity with wireless signal without explicitly specifying the wireless technology to be used [74, 253]. One approach using different communication channels but still relying on that communication being restricted to certain proximity is done in [35]. They propose two approaches: to use the magnetometer of the mobile device to sense a code of changes in the magnetic field caused by the token. Or to use the microphone of the mobile device to sense a acoustic transmission from the token. Both approaches again rely on the communication channel being restricted to certain proximity and could be extended by attackers with amplification or relays.

Interesting token choices have been made by Nicholson et al. [253], which were amongst the first to explicitly target (IBM Linux) wristwatches as tokens to automatically lock mobile devices when users depart. The advantages of using such a wristwatch as token are threefold – without the authors explicitly mentioning all of them: a) users already wearing a wristwatch are not required to think about carrying an additional token. b) it is less likely for wristwatches that are worn throughout a day to be lost or stolen, compared to tokens which might be unattended for certain times a day. c) wristwatches with computer functionality bring processing capabilities and interfaces required to perform token-based authentication to mobile devices. In case users already own such a wristwatch there would be no additional costs for acquisition of a token device. In a similar manner, Grosse and Upadhyay [128] mention a ring with NFC capabilities to authenticate to (mobile) devices. Similar to using a wristwatch as token this approach does not bear additional effort on users already wearing a ring, as they do not need to think about or carry an additional token device. Further, if the mobile device for usage is held with the hand wearing the ring, authentication effort would be small (only adjusting the grip of the device so that the NFC ring an NFC transponder of the device are close enough to perform authentication). The drawbacks of this approach are – besides authentication relying on the communication being possible only within certain proximity of mobile device and token – that rings with NFC functionality are unusual, thereby acquisition of the token is certainly connected to costs.

Summarizing, most token-based authentication approaches in the mobile environment share the majority of advantages and disadvantages. On the one hand, tokens are resistant to users choosing weak secrets, which would be the case with knowledge based authentication approaches, and they do not bear cognitive load on users to remember a secret. On the other hand, token acquisition might be connected to costs, tokens can be forgot, lost, or stolen. Consequently, they bear additional effort on users to carry the token along. Further, different token-based authentication systems might require users to remember and carry along multiple tokens. If tokens are lost or stolen, revocation is again connected to certain cost. Finally, tokens likely require extra time for performing authentication, e.g. by locating the token and presenting it to the mobile device to authenticate to. From an attacker's perspective token-based mobile device authentication could open up additional attack surfaces. These include the majority of approaches that rely on communication over wireless signals between token and device only being functional within a certain proximity. This assumption enables attackers to access the mobile device within this proximity (e.g. behind the user's back) or to extend the range of the signal to access it in a bigger distance. Other drawbacks include attackers being able to obtain the mobile device and the corresponding mobile token at the same time. This would enable attackers to easily perform authentication. Depending on how users secure their tokens this could be easier than shoulder surfing knowledge based secrets or capturing and spoofing the input to biometric based authentication.

Hence, though using tokens to authenticate to mobile devices has only been investigated by few previous research, there seem to be ways to unobtrusively use tokens in the mobile environment in the future. Many previous approaches accepted additional security issues for being unobtrusive, such as relying on wireless communication between token and mobile device only being functional in certain proximity. From this we conclude that there is room for new or additional ways of unobtrusive token-based user authentication to devices in the mobile domain.

3.5 UNOBTRUSIVE MIXED MULTI-MODAL MOBILE AUTHENTICATION

Authentication approaches can be combined to achieve improved authentication performance, either in the form of higher authentication security or as reduced obtrusiveness. The first includes e.g. multi-factor authentication where all factors need to be satisfied through successful authentication. For example, multi-factor authentication could combine two factors by using tokens with passwords, e.g. requiring users to present a token and enter a password, or to unlock a token using a password before it can be used to perform authentication [256]. Therefore, multi-factor authentication requires attackers to be in control or having obtained all factors, thereby leading to higher effort and costs to successfully perform attacks. However, multi-factor authentication also tends to cause increased authentication effort, as legitimate users are as well required to perform all individual authentication steps.

With the latter, combinations of authentication approaches can lead to reduced overall obtrusiveness for legitimate users [88]. While overall all types of authentication (knowledge, biometrics, tokens) could be combined, many approaches are designed to combine different biometrics. The main difference to multi-factor authentication is that usually not all authentication steps are required for successful authentication, but users could at any time choose the authentication they want to use, or systems could authenticate users implicitly depending on combinations of their actions, behavior, or alike. Therefore, usually a certain level of confidence that a legitimate user is trying to interact or is interacting with a device is required for successful authentication. Further, similarly to multi-factor authentication, including multiple modalities can also lead to better authentication results in terms of correct acceptance and rejection [10, 262].

Combining different authentication results often relies on fusion mechanisms. This can be achieved by fusion of original data or features (e.g. using multiple biometric modalities), authentication scores (applicable to all authentication approaches that yield a score), or authentication decisions [168, 293]. Different combinations of authentication approaches to achieve unobtrusive mobile authentication have been proposed in previous research. We subsequently discuss some interesting examples than use novel authentication aspects. One approach performs authentication during answering a call by sensing the device movement [10] or additionally by integrating the dynamics of how the slide swipe-to-unlock on the mobile device is performed, the arm movement to the ear, and voice recognition during the first 2.5 s of the call [51]. A related approach integrates users' micro hand movements of the first 10 s of device usage after unlocking a mobile device into explicit or implicit continuous mobile authentication [50].

Other examples integrate behavioral profiling, such as application usage [199] and device location history [10, 206, 207] and proximity to other devices/location fingerprinting. Face and touch modalities are combined in [316], while combinations of different keystroke dynamic approaches (during login and continuously during subsequent device usage) are combined in [100]. Another approach incorporating typing analysis combines text entered via the virtual keyboard with application usage profiles, visits of website, and physical location of the device (using GPS and/or WiFi) [111, 127]. In order to achieve unobtrusiveness such approaches often employ continuous authentication – which most frequently based on continuous biometrics [10, 262]. These approaches are frequently used in post-unlock manner, that is, after users performed regular authentication to unlock the device. Then unobtrusive authentication is used to further continuously authenticate users during subsequent device usage. Though investigating and integrating additional ways of continuous post-unlock authenticating is one important aspect for achieving overall unobtrusive mobile device authentication [9, 10], the initial authentication for unlocking a mobile device should be unobtrusive as well. This aspect has received less attention in previous research.

To enable generic combination of diverse authentication approaches different frameworks have been proposed and implemented. One exemplary and recent example would be CORMORANT [148, 152, 153], the Android framework for continuous, risk-aware multi-modal cross-device authentication. CORMORANT focuses on combining different authentication approaches across different mobile devices of the same users in a generic way. Further, current work with CORMORANT investigates the integration of the risk of mobile devices being physically accessed by third parties within their current context into the authentication decision[3]. For example, the likelihood of such access might be higher in public transportation than at home. Frameworks like CORMORANT aid development and integration of new authentication approaches by providing the surrounding framework that purposefully uses results and decisions of the underlying approaches. Therefore the development of a new authentication approach could focus on the approach itself, while the framework could take care of using and fusion of different authentication approaches and yielding an appropriate authentication decision on mobile devices.

Besides combining diverse authentication approaches to obtain unobtrusiveness, another important aspect is when to query users to perform explicit and thereby obtrusive authentication [9, 10, 88]. Approaches to this challenge often integrate implicit and/or continuous authentication to determine a suitable point in time for explicit authentication. Examples include a reduction of obtrusive authentica-

[3] The implementation of CORMORANT is currently ongoing with its source code being publicly available at https://github.com/mobilesec/cormorant.

tion to about 42% while obtaining 3.3%-16.1% false acceptance rate from the underlying implicit authentication [286]. This is achieved by combining multiple authentication approaches, including biometrics (face and voice), user behavior (changes in user behavior such as derived from different times and location of device usage), and token-based authentication (proximity to possessed nearby objects with BT/RFID signal strength). Another example uses text based continuous authentication to determine when explicit authentication is required [296]. They thereby combine linguistic text analysis, keystroke dynamics, and behavioral profiling.

Summarizing, with mobile authentication frameworks such as CORMORANT diverse authentication approaches can purposefully be combined and integrated on mobile device. Integration of diverse authentication approaches can thereby lead to robust authentication results (even when incorporating weak authentication approaches) while also leading to decreased overall obtrusiveness [10, 88, 262]. In this regard one aspect that remains open is the exploration and investigation of further novel approaches for unobtrusive mobile authentication. Thereby most previous work focuses on biometrics or behavioral aspects (e.g. profiling) while few approaches incorporate e.g. tokens into unobtrusive multi-modal mobile authentication. Furthermore, many approaches that rely on unobtrusive biometrics employ them in post-unlock manner, such as with continuous face authentication during device usage. This leads to the initial unlock potentially remaining obtrusive. Mobile authentication would thereby benefit from further unobtrusive biometric authentication approaches that can be utilized for an initial unlock. The employability of such approaches might be restricted to certain situations (such as gait authentication being restricted to users walking). Consequently, mobile authentication would again benefit from a diversity of approaches being available to perform authentication in different situations.

3.6 DEVICE-TO-USER AUTHENTICATION

User authentication with mobile devices usually assumes that authentication is done from users to their devices, e.g. to prevent unauthorized physical access to those devices. This form of authentication can be referred to as user-to-device authentication – but it is usually just referred to as user authentication due to it covering most of mobile device authentication involving users in literature. However, besides user-to-device authentication devices could also authenticate to their users with so called device-to-user (D2U) authentication. Little previous research addresses this form of authentication due to which it is practically unemployed on current mobile devices. This allows for hardware phishing attacks to be performed with most current mobile devices, which we discuss in the next section.

3.6.1 *Hardware Phishing Attacks*

When users start interacting with their mobile devices they implicitly assume the device they interact with to be the correct one. However, as devices usually do not authenticate to their users – in contrast to users authenticating to their mobile devices – it could also be an identically looking but different mobile device they are interacting with. This deception allows for what we refer to as hardware phishing attacks. At first attackers obtain an identically looking mobile device. They prepare it so that the same user-to-device authentication screen is shown. This screen is further prepared to relay every interaction with the phone to the attackers. This mobile device thereby becomes the phishing hardware, being an identically looking but malicious device that aims to deceive users into unwittingly revealing secret information to the wrong device. The attackers then exchange the user's mobile device with the phishing hardware while the user is inattentive. Subsequently, when users try to use their mobile device they at first authenticate – thereby revealing the authentication secret to the phishing hardware. The information is relayed to the attackers who can use it to authenticate to the device previously obtained from the user and unlock it. The reason why we refer to these attacks as hardware phishing attacks and to the devices simply as phishing hardware is that these work by deceiving users in the same manner as e.g. web-site based phishing attacks.

As with all phishing attacks – including mobile hardware phishing attacks – the malicious instance just needs to mock the real instance until authentication credentials have been revealed. It is already too late if users recognize moments later that they are interacting with a wrong device – especially, if the real device is already out of their reach. In contrast to web-based phishing, after users recognize a hardware phishing attack is ongoing, the legitimate device is (and most likely stays) under control of attackers. Therefore, while hardware phishing attacks have a higher initial cost than their web-based counterparts (as new hardware is most certainly required for each attack), the cost is not lost during the attack. Attackers could reuse or sell the acquired device after an successful attack and thorough analysis of data on the device. Although the initial cost might affect the cost-to-gain ratio to be too high for some targets, for other targets hardware phishing attacks would certainly still be profitable – e.g. for obtaining business or industry intelligence. Additional issues are that a) virtually all mobile device models are strongly standardized (including possible customizations in software and look-and-feel), and identical copies of all these models can be easily obtained by attackers. b) hardware phishing attacks cause devices to be swapped – hence for attackers there is no loss in terms of hardware. and c) individual/personal customization (e.g. screen wall paper, sounds, even hardware customizations as stickers on the device) could as well be duplicated easily by attackers for the mock device. Obtaining information about the target device and its features for creating phishing hardware can be done by attackers without physical access to the device, e.g. by inconspicuously taking pictures of the phone (e.g. while it is lying on a table).

Mobile devices authenticating to their users (as users do to mobile devices) would be an effective measure against such attacks. This could be done e.g. by revealing a shared secret to users, so that they are assured that the device is in fact the correct one.

3.6.2 *Previous Approaches to Device-to-User Authentication*

Little previous research has focused on D2U authentication. One approach to D2U authentication is by devices visually revealing secret information to users to ensure they can be trusted. An example for this are web-based banking systems where after logging in users are presented a previously defined secret to ensure authenticity of the service they are interacting with. Another example is displaying variations of secret images to the user to assure authenticity of user interfaces and computer systems [288, 289]. The main drawback of such approaches is being prone to shoulder surfing attacks (an attacker visually observing secret information revealed to the user by the device – without requiring physical access to the device).

Other related approaches deal with human verifiable authentication when pairing devices (e.g. Bluetooth pairing in general [145]) or pairing of devices with restricted in- and output capabilities (e.g. pressing a button on device A in the same pattern a LED blinks on device B [202] or shaking devices together [224]). In contrast to these mechanisms which are intended to be employed once during device pairing (hence, reduced usability is experienced only once and the risk of e.g. being shoulder surfed can be avoided by additional effort), D2U authentication is intended to be used frequently. Consequently, usability drawbacks through additional effort would impact users more frequently.

The reason for D2U authentication being employed rarely can be explained with a comparison to mutual authentication between machines, as both users of mobile devices authenticate to their devices and devices authenticate to their users this can be considered to be a form of mutual authentication. While mutual authentication is well established in machine-to-machine (M2M) communication (e.g. web technologies like IPsec [83]) it is rarely used for authentication involving humans. This is because in contrast to M2M authentication, both U2D and D2U authentication are limited by certain human factors. In comparison to M2M authentication, D2U authentication is especially limited in channel bandwidth (exchange of larger portions of information takes longer for humans than computers) and computational capabilities (e.g. cryptographic mathematics, which humans can hardly do without aid of computers). Both make communicating authentication information from devices to humans more challenging than communicating it in between machines. D2U authentication is further limited by previously discussed additional cognitive load and time to perform authentication in the same manner as U2D authentication.

As little previous work covers mobile D2U authentication this field is still open for research and proposals of novel approaches. Similarly to previously discussed security mechanisms employing these approaches might result in a trade-off between security and usability where increasing security tends to decreases usability and vice versa [72]. Consequently, D2U authentication approaches need to be designed carefully, with their obtrusiveness in mind and as little as possible overhead for users. However, as D2U authentication is covered by little previous research, even approaches focusing on being unobtrusive at the cost of proving less-than-optimal security will lead to a security gain on mobile devices.

3.7 SUMMARY

To protect data on mobile devices from unauthorized physical access of third parties, different concepts of mobile authentication can be employed. However, as discussed in this chapter, employing authentication usually comes at the cost of also impeding daily usage of mobile devices by legitimate users. The resulting trade-off between security and obtrusiveness is apparent for all three different types of authentication: knowledge, inherence (biometrics), and possession (tokens). To summarize authentications concepts presented in related work we attempt to categorize their most important characteristics as being either advantageous, neutral/variable/not applicable, or disadvantageous in terms of mobile environments (Tab. 1).

The most frequently employed knowledge based authentication approaches include PINs and passwords on desktop computers as well as PINs and graphical patterns on current mobile devices. Their core advantage is that the authentication secret can be changed easily in case it is being disclosed to third parties. Their drawbacks include increased cognitive load and additional time required to perform authentication, especially on mobile devices. As a result this leads to users choosing weak knowledge based authentication secrets as well as some users not using knowledge based authentication approaches at all. Graphical passwords, of which graphical patterns are a special form, have been designed to reduce the corresponding cognitive load imposed on users. However, they are by design unable to prevent cognitive load altogether. This is especially problematic when scalability

Modality	Type	Cognitive load	Weak secrets	Exchangeability	Authentication duration	Input cumbersome	Additional costs	Shoulder surfing	Smudge attacks	Hardware phishing attacks	Remarks
PINs	U2D, knowledge	-	-	+	+	-	+	-	+	-	–
Passwords	U2D, knowledge	-	-	+	-	-	+	-	+	-	Slower than PIN, higher cognitive load
Graphical passwords	U2D, knowledge	-	-	+	-	-	+	-	-	-	Cognitive load smaller than with PIN and PW, slower than PIN
Graphical patterns	U2D, knowledge	-	-	+		-	+	-	-	-	Faster than most graphical passwords, slower than PIN
Various biometrics	U2D, inherence	+	+	-				+	+	-	Need additional measures to protect biometrics, applicability situation dependent, can provide for transparent authentication
Tokens	U2D, possession	+	+				-	+	+	-	Additional costs, additional piece of HW, proximity based approaches could be exploited by attackers
Multi-modal (multi-factor)	U2D, multiple				-	-				-	Higher security, more obtrusive
Multi-modal (multiple options)	U2D, multiple				+	+				-	Less obtrusive, weakest modality determines system security
D2U	D2U, knowledge	-	-	+		+				+	At the time being only considering knowledge: countermeasure to hardware phishing attacks, no mobile approaches available for reference

Table 1: Overview of authentication concepts in related work, with different user-to-device (U2D) and device-to-user (D2U) authentication modalities. Tendencies of the most important characteristics within mobile environments derived from related work are attempted to be summarized as advantageous (+), neutral/variable/not applicable (empty), or disadvantageous (-).

becomes important with multiple mobile devices, frequent device usage, and the requirement to use complex but different authentication secrets for different devices.

In contrast, biometric authentication does not bear additional cognitive load on users for remembering an authentication secret. However, unlike knowledge based secrets, biometrics cannot easily be changed in case they are disclosed to third parties. As a consequence, while mobile biometric authentication is less obtrusive, it exposes their users to the additional risk of biometrics being disclosed. This is why biometrics deserve adequate protection on mobile devices with e.g. algorithmic template protection or secure hardware. While template protection relies on the algorithmic security of the corresponding approaches and can be computationally expensive, approaches using secure hardware like smart cards have to be designed with corresponding computational limitations in mind. As a result, existing approaches are strongly adapted to individual biometrics and can usually not be applied generically to the diversity of biometrics available to modern mobile devices. Therefore, to utilize various biometrics on mobile devices there would be a need for mobile biometric authentication approaches that are suitable for secure hardware while also being generically applicable to different biometrics.

Similarly to biometrics, token-based authentication prevents users from choosing weak secrets and does not impose cognitive load on users for remembering an authentication secret. However, it imposes additional cognitive load in requiring users to remember to bring the token along and to have it available for authentication. As tokens are likely required to be as mobile as users' mobile devices they can as easily be forgotten, lost, or stolen as the mobile devices themselves. To perform authentication the token needs to be presented to the mobile device in some way – which can be obtrusive for users and requires additional time. Depending on the authentication mechanisms this can also result in the authentication being easily circumventable by attackers if they are able to e.g. access the device while it is in proximity to both its owner and the authentication token. Additional complications with token-based mobile authentication arise from costs to purchase and revoke tokens, potentially different tokens being required for different authentication approaches, and the requirement to bring these different tokens along with mobile devices. However, there seems to be room for novel mobile token-based authentication approaches that utilize multiple mobile devices so that e.g. one device becomes the token for authentication to other devices.

The combination of multiple authentication approaches on mobile devices, possibly incorporating different modalities, seems promising. One advantage of such combinations is that individual approaches can better focus on a subset of situations in which authentication is required than one single approach that would need to cover all those situations. Combining different authentication approaches gives users choices to use the best suited approach in a certain situation, therefore has the potential to reduce overall obtrusiveness of mobile authentication. If implicit authentication approaches are incorporated this could further lead to users being authenticated transparently in certain situations, e.g. when a smart phone is in the trousers pocket while walking. Frameworks for mobile authentication like CORMORANT thereby facilitate the integration of diverse and novel authentication approaches. This is because developers are able to focus on the corresponding authentication approach and can leave the utilization of its result (i.e. aggregating authentication from different modalities and deriving an overall authentication decision) to the framework. The development of additional, alternative, and novel mobile authentication approaches is further aided by the increasing amount of data mobile devices have access to. For example, mobile authentication could incorporate many different sensed biometrics ranging from ECG [25] to wrist vein authentication [101], where each could suit different authentication situations. The more such authentication approaches are available, the more options users have to choose from, hence the higher the chances that one approach will suit the current situation and provide for an improved user experience.

In contrast to user-to-device authentication, mobile device-to-user authentication is rarely addressed with existing literature. This potentially enables attackers to perform hardware phishing attacks with current mobile devices – but also leaves room for novel approaches and proposals of how mobile devices could authenticate to their users. However, similar to regular user-to-device authentication these approaches need to be designed with their obtrusiveness in mind, including cognitive load and additional time required to perform authentication.

To summarize, these issues illustrate that a number of areas and challenges with mobile authentication remain interesting for future research. These especially include intensified usage of diverse mobile biometrics together with according mechanisms to protect used biometrics, additional and alternative authentication approaches that, amongst others, incorporate multiple mobile devices of the same user, as well as the combination of diverse authentication approaches to better suit diverse mobile authentication situations. In those areas novel unobtrusive approaches could contribute to improving the overall user experience of mobile authentication, facilitate their usage, and thereby reduce the effectively applicable threat model of unauthorized physical third party access to mobile device data.

Part II

OUR APPROACH: UNOBTRUSIVE MUTUAL MOBILE AUTHENTICATION WITH BIOMETRICS AND MOBILE DEVICE MOTION

4 OUR APPROACH: AN OVERVIEW

Our work resides in the field of physical access protection of devices in the mobile environment. As discussed in the last sections and chapters, existing approaches to protect mobile devices from unauthorized physical access of third parties are sometimes not used, or not used to their full capacity due to additional effort imposed on users. This leads to those devices – thereby the data processed and stored on them – being accessible by unauthorized people in multiple situations throughout daily device usage. In our work we aim for providing additional physical access protection mechanisms for mobile devices. We aim to not bear significant additional effort on users and to suit the diverse situations in which authentication might be required. Overall, our approach thereby follows these statements: the less classic, obtrusive and explicit authentication is required and the more unobtrusive the core mechanics of employed approaches are, the lower the overall obtrusiveness of mobile authentication becomes, and the more realistically authentication is actually employed by end users in daily device usage [9, 10, 88]. Thereby, the more different unobtrusive authentication approaches are available to users (e.g. authentication being possible in one of multiple ways) the higher the chance that one such option is suitable for the current situation and that it bears little or no additional overhead on users [109, 262].

Consequently, with our approach we enable new ways of unobtrusively performing authentication with mobile devices by incorporating both user-to-device and device-to-user authentication (Fig. 10). Our user-to-device authentication consists of an approach to generic biometric MOC authentication and a token-based approach to unobtrusively transfer the authentication states between mobile devices utilizing mobile device sensors and device motion. Our device-to-user authentication utilizes vibrations to communicate an authentication secrets to users. We shortly introduce these approaches in the subsequent sections and discuss them in depth in their corresponding chapters (Cha. 5, 6, and 7).

4.1 USER-TO-DEVICE AUTHENTICATION

With user-to-device authentication we strive for less obtrusive approaches and providing additional ways of users authenticating to their devices. We further aim to protect users' biometric data used for authentication on mobile devices from being disclosed to third parties or transferred to devices outside the control of users. We ad-

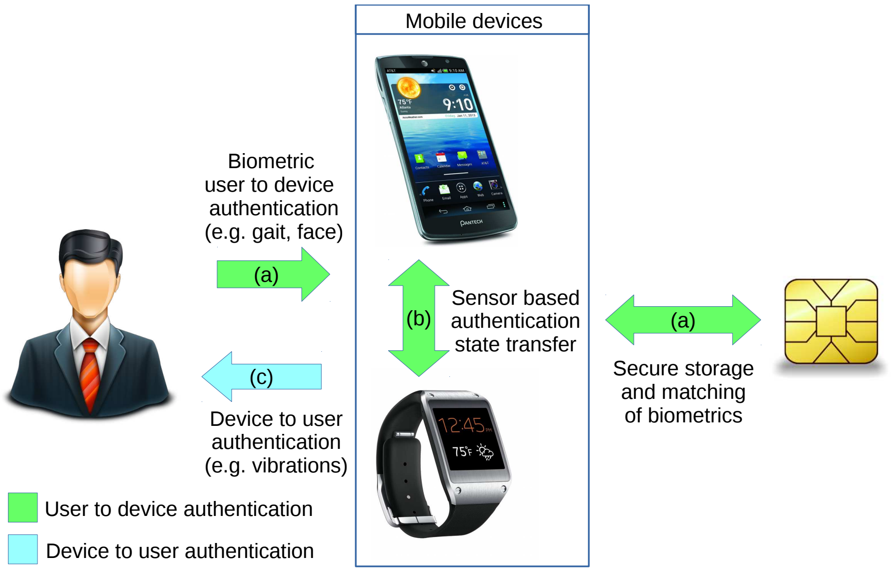

Mobile devices
Biometric user to device authentication (e.g. gait, face)
(a)
(c)
Device to user authentication (e.g. vibrations)
(b)
Sensor based authentication state transfer
(a)
Secure storage and matching of biometrics
🟩 User to device authentication
🟦 Device to user authentication

Figure 10: In our approach we incorporate user-to-device authentication as (a) biometrics based MOC and (b) sensor based token authentication, and (c) vibration based device-to-user authentication.

dress those goals with an open, transparent, and generic biometric MOC authentication to protect users' biometrics and sensor based token authentication to easily transfer authentication states between devices to unlock them.

4.1.1 *Biometric Authentication: MOC Authentication for Multiple Biometrics*

For securely using biometrics with mobile user-to-device authentication we present a MOC approach that is applicable to multiple biometrics in a generic way (Fig. 10, a). Our approach uses a training dataset for a specific biometric and offline training (outside mobile devices, e.g. on desktop or server hardware) to obtain an authentication model with a simplistic internal representation in the final trained state. We then adapt and simplify features and model representation to enable their usage on SCs.

Generic biometric MOC authentication thereby bears several advantages with respect to the stated goals of user-to-device authentication. Using biometrics for user-to-device authentication bears no cognitive load and be preformed easily and quickly. The exact effort and duration can be influenced by the choice of biometrics and the employed sensing mechanisms. Depending on type of biometrics, authentication can either be explicit (users being aware of authentication going on, e.g. with classic explicit face or fingerprint authentication) or implicit (users not being aware of authentication going on, e.g. possible with continuous gait or face authentication). Overall obtrusiveness can be reduced with the latter or combining both types. The obtained model can be used within SCs on mobile devices without requiring retraining when enrolling new users. This leads to users not being required to download any data to their mobile devices that represent negative class samples during training. Further, the lengthy and battery draining model training process itself is not required on mobile devices at all. Enrollment just requires the storage of samples from the user. Biometrics are stored on mobile device within SCs. They cannot easily be read from storage even if the device comes under control of attackers. Attackers would need to be in control of the device and be able to monitor device memory while legitimate users enroll or authenticate using their biometrics and our approach. This raises the bar for disclosure of biometric information by increasing the effort required for attackers. Finally, we argue that this approach being generic can aid transition of other biometrics to using MOC authentication in the future.

4.1.2 *Token Authentication: Transferring Authentication States Between Devices to Unlock Them*

For reducing the number of times classic user-to-device authentication is required, and to add another option to performing user-to-device authentication, we present ShakeUnlock, a token-based mobile device unlocking approach based on briefly shaking two devices conjointly (Fig. 10, b). We transfer the authentication state from the already unlocked device to the locked device to unlock it as well. A common use case would feature a wrist watch as token device, which remains unlocked as long as it is strapped to the user's wrist, and a locked mobile phone, which is unlocked after both devices are shaken conjointly.

Transferring authentication states to unlock mobile devices thereby bears several advantages with respect to the stated goals of user-to-device authentication. Shaking does not cause additional cognitive load and requires little user attention to be performed. Users do not have to look at the device during shaking which can be performed single-handedly. This allows for unlocks e.g. while walking and carrying a bag with the other hand, and without looking at the device screen. Providing shaking as alternative token authentication method can reduce the number of times knowledge based or biometric authentication is required to unlock mobile devices. Forging shaking patterns is difficult, which impedes malicious unlocks in case attackers gain control over a locked device but not the corresponding token device.

4.2 DEVICE-TO-USER AUTHENTICATION

To let mobile devices communicate authentication information back to users, we present vibration based device-to-user authentication (Fig. 10, c). This represents a countermeasure to hardware phishing attacks, in which attackers replace the device with an identical-looking malicious device to eavesdrop on users revealing their authentication secret to the device. The revealed authentication secret could be transmitted to the attackers immediately, who then conveniently authenticate to the real device. To impede such hardware phishing attacks we let devices communicate an authentication secret back to users in parallel to them authenticating to their devices.

This approach to device-to-user authentication bears several advantages with regard to our stated goals. It can be performed without requiring additional authentication time if performed in parallel to users authenticating to their devices. Users might further become familiar with their pattern, similar to being able to type a password using muscle memory with user-to-device authentication. While we are not aware of any studies on a muscle-memory-like effects on intuitively recognizing vibration patterns, from previous studies on muscle-memory effects [16, 205, 307, 311, 323, 370] we conjecture that such effects could also be possible with vibration patterns. This would allow users to intuitively recognize that "something changed" in case of the pattern being different without significant additional effort. In this case users can stop user-to-device authentication going in parallel to not fully reveal their authentication secret. Further, observing device-to-user authentication information communicated via vibration is more difficult to observe for attackers e.g. using a visual or audio channel, which impedes eavesdropping attacks on this information.

4.3 COLLABORATION OF CONSTITUENT PARTS WITHIN OUR APPROACH

By incorporating our improvements to mobile biometric and sensor based authentication we reduce the overall effort users need to dedicate to authentication related tasks for physical access protection on mobile devices. Authentication tasks become overall less obtrusive: users can be authenticated by their mobile devices with implicit and possibly continuous biometric authentication approaches without even noticing it, which does not bear any additional authentication effort on users. As those biometric authentication approaches use MOC techniques to protect involved information, theft of biometrics becomes noticeably more difficult for attackers. In cases where implicit authentication is not suitable we can still offer multiple ways of performing user-to-device authentication, namely a) explicit biometric authentication, again using MOC techniques, and b) token-based authentication by shaking devices conjointly. For example, to one handedly unlock a mobile phone they just picked up users could choose to either use e.g. face authentication or briefly shake the phone conjointly with the already unlocked smart watch strapped to their wrist. The common advantage of those approaches over frequently used, classic knowledge based authentication approaches is that they do not bear cognitive load on users. Further, by providing multiple ways of authentication for one situation, chances are higher that one way is well suited for the situation and only implies little overhead for users.

In all those authentication situations devices can perform device-to-user authentication too, by communicating an authentication secret back to users. This addresses hardware phishing attacks by raising the effort required to trick users into authenticating to the wrong devices. Such device-to-user authentication bears little additional effort on users, as it can be done in parallel to users authenticating to devices themselves.

Note that while our approach will not be suitable to fully replace classic, knowledge based authentication approaches, it is meant to aid mobile authentication by reducing the number of times classic authentication is required. The aim is to make authentication overall more manageable with multitudes of mobile devices. In cases where our unobtrusive authentication approaches are unsuitable, classic authentication is meant to be used as fallback. Overall, we argue that with our approach we thereby contribute to advancing mobile authentication and provide one further step towards making authentication with a multitude of personal mobile devices unobtrusive and manageable.

4.4 A PERSPECTIVE ON OUR APPROACH IN A WIDER CONTEXT

Our work is done in relation to and in corporation with CORMORANT [148, 152, 153], the Android framework for continuous, risk-aware multi-modal cross-device authentication. This framework is in the focus area of a separate PhD topic driven by Daniel Hintze, ongoing in parallel to this present thesis at the Institute for Networks and Security (INS), Johannes Kepler University (JKU) Linz, Austria. CORMORANT focuses on combining different authentication approaches across different devices of the same users in a generic way. The implementation of CORMORANT is currently ongoing with its source code being publicly available[1].

From the perspective of CORMORANT our work can be seen as ground laying work that provides additional approaches for unobtru-

[1] CORMORANT framework source code: https://github.com/mobilesec/cormorant.

sive authentication on mobile devices. In contrast to CORMORANT, it thereby focuses only on one to at maximum two devices at the same time. Further, our work does not focus on fusing authentication results from different authentication procedures possibly conducted in parallel on mobile devices. However, our approach can be embedded in CORMORANT in the form of authentication modules[2]. It thereby contributes to a diverse ecosystem of authentication approaches that can be used across multiple mobile devices, with multiple authentication procedures possibly ongoing in parallel, and without bearing additional authentication effort on users.

[2] See https://github.com/mobilesec for details on individual CORMORANT authentication modules.

5 A GENERIC APPROACH TO MOBILE BIOMETRIC MATCH-ON-CARD AUTHENTICATION

In this chapter we highlight our biometrics based user-to-device authentication approach. It is applicable to different biometrics and uses offline training with feature and model simplification to enable the usage of features and models on SCs (Fig. 10, a). Parts of this chapter have previously been published in [103, 104].

With modern mobile devices and their many different sensing capabilities it is reasonable to employ multiple biometrics in order to achieve unobtrusive authentication in different situations. However, biometrics need to be protected accordingly as they cannot easily be changed in case of disclosure (Sec. 3.3). With modern mobile devices and modern SIM cards it is reasonable to use smart cards (SC) to protect biometrics used for authentication. The reason for this is that modern mobile devices and SIM cards start featuring built-in SCs – which cancels out the otherwise additional purchase cost of secure hardware to protect biometrics. However, designing approaches for biometric authentication utilizing SCs is challenging due to the computational limitations of SCs (Sec. 3.3.3). Approaches thereby have to be designed so that they are feasible within the limited storage and processing capabilities of SCs. These limitations of SCs affect both the internal structure of authentication models and number and type of features that can be used with SCs. Further, as the transmission bandwidth to/from SCs is limited, the amount of data that can reasonably be sent to SCs during user authentication is limited as well. In order for operations with biometric TOC and MOC approaches to be feasible on SCs, the used operations and approaches are usually domain specific. This impedes approaches being applied to different biometrics as the underlying operations have to be adapted accordingly.

To address these restrictions we aim for enabling a more generic usage of simple machine learning (ML) models on SCs. Our generic MOC approach computes authentication models offline with sufficient computational power and does not require the models to be retrained during enrollment of individual users. The challenge therein lies with the mentioned limitations of SCs which imply restrictions in how biometric features and ML models can be calculated and represented for usage on SCs. We therefore propose a scheme which trains and generates ML models offline (e.g. using server infrastructure), then uses the simplified internal structure of trained models on SCs in the matching process (Fig. 11).

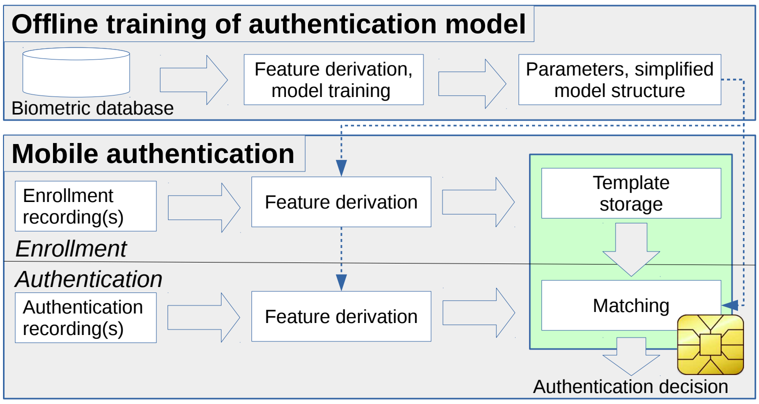


Figure 11: Conceptual overview of our MOC approach. The SC is highlighted in green.

Models suitable for this approach are those where the internal structure translates to a simple representation in the final and fully trained state (e.g. an equation). In contrast to matching on the SC, the offline training, evaluation, and selection necessary to obtain this structure in the first place can be arbitrarily complex. After obtaining such a model offline, both features and models need to be adapted to suit SC restrictions. This includes data types of features and models, as well as computations using those. Note that it is desirable to integrate necessary adaption to features and models already in the offline modeling process. Doing so allows for more precise estimation of authentication performance, which is in turn important for model tuning and selecting a reasonable model and model configuration for usage on SCs. Consequently, both offline and on-device processing rely on identical preprocessing and feature derivation. Further, note that feature derivation up to feature simplification can be performed outside the SC. This allows for more complex and powerful feature derivation while not compromising any information previously stored on the SC.

We demonstrate our generic MOC approach on acceleration based gait biometrics as well as face biometrics, using SCs restricted to either 16 or 32 bit range integer calculations. We transform features derived from biometric recordings and model structure used on the SC to be represented in half of the integer range available on the SC. This allows for multiplications within the available integer range. We demonstrate that adequate MOC authentication is still feasible using limited bit representation of the obtained model, stored biometric template, and new biometric recording. Summarizing, the contributions of our biometric MOC authentication approach are:

- We present a generic approach towards biometric MOC authentication, wherefore we adapt both offline trained ML models and features to enable their computation and handling on SCs.

- We apply our generic MOC authentication approach to face authentication and acceleration based gait authentication as examples of biometrics with usually more complex matching and bigger templates. To the best of our knowledge, this is the first practical approach to gait MOC authentication with acceleration data.
- We evaluate the feasibility and performance of our generic MOC authentication approach with publicly available data sets, using both 16 and 32 bit Java Card SCs. We achieve 11.4% and 2.4-5.4% EER for gait respectively face authentication, while staying in the range of 2 s respectively 1 s for transmission and calculation durations on SCs.

Our approach to generic mobile biometric MOC authentication thereby facilitates the secure usage of different biometrics with mobile authentication. This can further facilitate more biometrics being transferred to using MOC techniques, thereby more biometric authentication approaches (that enable unobtrusive authentication in different situations) being available to mobile users without exposing them to the additional risk of disclosing their biometrics.

5.1 RELATION TO PREVIOUS FACE AND GAIT MOC AUTHENTICATION

The reasons for choosing gait and face biometrics for the evaluation of our generic MOC approach are twofold. Firstly, both biometrics can be utilized on most modern mobile devices as their recording only requires cameras and acceleration sensors. Both are shipped with most modern off-the-shelf mobile devices, hence enabling the usage of both gait and face biometrics on most mobile devices. Secondly, the differences between gait and face biometrics emphasize the applicability our MOC authentication approach to different types of biometrics. Both are representative for different types of biometrics. Gait represents behavioral, weak, and continuous biometrics. It can be used to unobtrusively authenticate mobile users while walking. While the applicability of gait authentication is limited to the duration of users walking, it virtually requires no user attention for authentication. In contrast, face authentication represents physiological, strong, and non-continuous biometrics. While face authentication can be considered to be more obtrusive, its authentication confidence is stronger and its applicability is arguably wider than with gait biometrics. Further, different underlying features and matching approaches are usually employed with face and gait biometrics.

To the best of our knowledge, there exist no previous approaches to acceleration based gait MOC authentication. With the majority of existing gait authentication approaches (Sec. 3.2.4), either retraining the model for individual users would be required, or neither training the model, nor using a ready trained model to predict new samples is feasible on SCs with respect to their computation requirements. This specifically concerns approaches using DTW during matching templates. For two time series of length m and n, DTW brings a memory complexity of at minimum $m \cdot n$, which renders it infeasible for usage on regular SCs. Though there exist some effective approaches to reduce the computational complexity of DTW (thereby also restricting its warping power), such as the Sakaboi-Chiba band [255, 297], even most limited DTW approaches are difficult to calculate on SCs. Consequently, SC based gait authentication has to utilize different types of models for matching templates. Still, feature derivation mechanisms used in gait authentication literature can be adapted for gait MOC approaches – as long as it is computed outside the SC.

The work closest to our MOC approach applied to face biometrics is Bourlai et al. [38]. Commonalities include the usage of an LDA model, a linear combination, and a threshold for the authentication decision. Still, both approaches rely on different core mechanisms: a) we do not use samples such as faces directly, but distances between samples to distinguish between comparisons of samples of the same person from those of different people. As we only train our model once offline, we can ship the pre-trained model with SCs on mobile devices. This allows enrolling new users without requiring any retraining, while the enrollment of one user is still completely independent of the enrollment of other users. b) with a client specific LDA, the distance to the client template is combined with the distance to the mean of impostors in a one dimensional way. In contrast, we use our model and multi-dimensional distances between a new sample and the reference template to derive an authentication decision. c) we perform feature derivation outside the SC. This prevents computing features for the enrolled template on the SC for each authentication attempt as done in [38] and allows for computationally more intensive operations during feature derivation in general. The downside is that this prevents exchanging feature derivation for existing templates at a later point in time. In summary, our MOC approach utilizes the distances between samples to distinguish between comparisons of samples from the same person and those of different people. In contrast to previous work on face MOC authentication, we can further ship the pre-trained model with SCs on mobile devices without requiring any retraining to enroll of users.

5.2 THREAT MODEL

When biometrics are used for authentication on mobile devices attackers could strive to compromise users' biometrics as well as to circumvent the authentication using the obtained biometric data. Attackers could try to obtain data about users' biometrics outside mobile devices. This could be done by collecting publicly available biometric data, by gaining unauthorized access to confidential computers, services, or databases that store and/or process biometrics data, or by attackers recording biometric data of legitimate users themselves. As our MOC approach does not focus on protecting biometric data outside mobile devices those attack vectors are declared out of scope.

On mobile devices, attackers could try to compromise a biometric system that is not protecting biometric data on different ways (Fig. 12) [274].

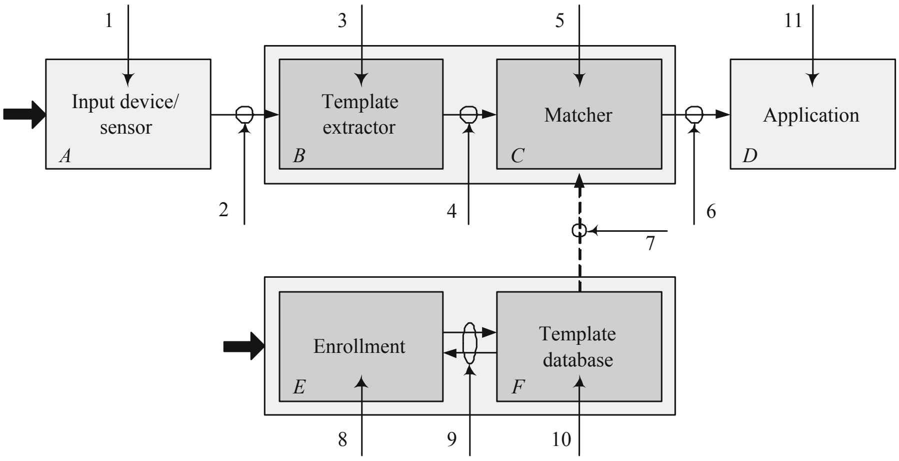


Figure 12: Attack vectors to biometric systems not protecting biometrics [274].

Attack vector 1 refers to attackers presenting fake biometrics to the sensor (to achieve authentication) and eavesdropping biometric data recorded by the sensor (to compromise users' biometrics). Attack vector 2 refers to eavesdropping or manipulating the communication from sensors to the authentication software. Attack vector 3, 4, and 5 include eavesdropping or manipulating the feature extractor, the matcher, or the communication in between them. Attack vector 6 refers to manipulating the authentication decision to achieve authentication. Attack vector 7 refers to eavesdropping or manipulating the communication between the template storage and the matcher. Attack vector 8 refers to eavesdropping or manipulating the enrollment or enrollment data. Attack vector 9 refers to eavesdropping or manipulating the communication between enrollment and the template storage. Attack vector 10 refers to extracting or manipulating stored biometric data. Attack vector 11 refers to attacking the application that utilizes the authentication decision.

With our MOC approach we strive to protect biometric data used for authentication on mobile devices from disclosure to third parties. Biometric data is involved in attack vector 1-5 and 7-10, which attackers could therefore use to extract biometric data about the legitimate user. However, the required capabilities of attackers and the required timing for those attacks differ. With attack vector 10, after the legitimate user has enrolled, attackers could access the template storage by different means. Besides others, these include bringing the device under their physical control and accessing the storage via a file system as well as physically disconnecting the storage from the mobile device and connecting it to custom hardware to read the information it contains. These attacks neither require attackers to be able to live monitor or manipulate device memory (e.g. using malware executed with elevated privileges on the mobile device) nor do they need to be performed at a certain time.

With capabilities to monitor or manipulate the device memory attackers might also be able to extract biometric data using attack vector 1-5 and 7-9. They could thereby directly read sensor values or eavesdrop biometric data between sensor and feature extraction, between feature extraction and matcher, between enrollment and template storage, or between template storage and matcher. They could further monitor the computations done for feature extraction, matching, or enrollment to extract biometric data. An important difference between those attacks is the timing when they are possible. To extract biometric data attack vector 1-4 can only be exploited while the legitimate user authenticates, and attack vector 8-9 only while the legitimate users enrolls. In terms of malware, this requires attackers to run such malware while the legitimate users enrolls or authenticates for extraction of biometric data to be successful. In contrast, attack vector 5 and 7 could also be exploited while an authentication attempt is made, independently of it being made by the legitimate user or not. This enables attackers to extract biometric data without requiring any interaction by the legitimate user (e.g. by bring a device under their control, manipulating it, and triggering an authentication attempt). Consequently, the most important weaknesses of biometric systems not protecting biometric data are attack vector 10, 7, and 5 – which are the attack vectors our approach addresses using MOC techniques.

Attack vector 10 is addressed by using a TOC approach for storing biometric data. With TOC approaches, attackers cannot access data in the template storage and are required to trigger an authentication attempt for templates to be fetched from the SC[1]. Attack vector 5 and 7 are addressed with using a MOC approach – like the one we propose – instead of a TOC approach. As with MOC approaches biometric data stored on a SC never leaves it attackers can neither access the communication between template storage and matcher nor the matcher itself. Attackers capable of monitoring device memory are therefore required to perform eavesdropping to obtain biometric data while the legitimate user enrolls or authenticates.

[1] Attacks on the security of SCs themselves, such as side-channel attacks by Kocher et al. [182] or Vermoen et al. [354], which try to extract the biometric template from the SC itself, are defined to be out of scope.

In order to address attack vector 1-4 and 8-9 for attackers capable of live monitoring device memory while the legitimate user enrolls or authenticates, securing/hardening the whole processing chain from sensors up to the authentication decision is required. This also addresses another form of attack within attack vector 2. Attackers which can physically manipulate the mobile device could add additional eavesdropping hardware in between the sensor and the feature derivation. This would enable them to eavesdrop sensed biometric data without requiring capabilities to live monitor device memory. One approach to protect the whole processing chain from the sensor to the authentication decision is to combine MOC with a trusted execution environment (TEE, e.g. ARM TrustZone[2]) that protects information from sensors up to the SC. Another approach is to combine all steps in an all-in-one piece of hardware, which is referred to as system-on-card (SOC), and of which MOC represents the essential part of internally matching biometric samples.

This is why both the combination of MOC with a TEE as well as SOC can be seen as a superset of MOC. Consequently, providing generic and widely applicable mobile MOC approaches is an essential part of fully protecting biometric information on mobile devices from attackers with live eavesdropping capabilities. Our approach towards generic MOC authentication is a first step towards the long-term goal of protecting mobile biometrics in a transparent and well evaluated way. For the first time it combines a MOC approach, generic matching concepts, and biometrics with traditionally bigger, therefore more challenging templates (such as facial images and gait cycles compared to e.g. fingerprints). This is why we purely focus on the MOC aspect and, for the time being, declare other attack vectors, such as the usage of malicious software/trojans on the sensor data processing pipeline to be out of scope.

5.3 GENERIC BIOMETRIC MOC AUTHENTICATION

Our MOC approach is divided into offline model generation and usage of the obtained model for enrollment and authentication on the mobile device. Both parts share steps for preprocessing, feature derivation, and feature simplification (Fig. 13). The offline part determines the parametrization which is then applied on mobile devices alike. On the mobile device those steps are done outside the SC, which thereby allows for computationally more complex operations or operations specific to certain biometrics. Based on preprocessed biometric samples, offline computation trains an authentication model, simplifies it, applies feature selection, and finally estimates the resulting authentication performance. The obtained model

[2] ARM Trust-Zone: http://www.arm.com/products/processors/technologies/trustzone/

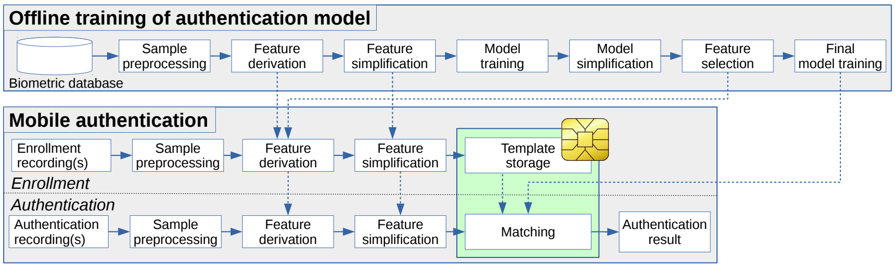


Figure 13: The offline part of our generic MOC approach computes and simplifies an authentication model, then selects the most important features to be used on mobile devices. On mobile devices, our approach uses the determined parameters and model to perform MOC authentication. The SC is highlighted in green.

is stored on the SC integrated in mobile devices, which then performs the MOC operation using stored samples and newly recorded samples. Therefore, no (re)training of the model is required in order to enroll new users.

5.3.1 *Offline Model Creation*

With X bit SCs, integer operations within X bit range are done in hardware, therefore are fast. We consequently strive to keep computations on SCs within this range. More specifically, we use a linear model on the SC, which internally computes a result using a linear combination of feature vector and model slope vector[3]. We therefore adapt features and model slope so that their linear combination is possible within X bit range on the SC.

On the one hand those simplifications lead to faster computations. On the other hand they also lead to a more coarse resolution of the feature space. For example: the feature space of 10 features expressed in 8 bit is limited to $2^{8^{10}} \simeq 1.21 \cdot 10^{24}$ possibilities, which corresponds to a theoretical maximum entropy of 80 bits. Expressing the same features in 16 bit results in twice the theoretical maximum entropy of 160 bits[4]. One could assume that using less information in features and models (due to using 16 instead of 32 bit SCs) would reduce the subsequent authentication accuracy. However, our evaluation indicates the impact to be negligible.

[3] The slope is a vector of numeric coefficients and defines the direction and steepness of linear models.

[4] Due to the uneven distribution of biometrics in feature space, biometric approaches are usually unable to exploit the full feature space [248]. Hence, depending on the used biometrics and features, the resulting true entropy is necessarily smaller than this theoretical boundary.

5.3.1.1 *Feature Simplification*

To work with X bit integer space SCs, we transform (scale, shift, and round) original real-valued features to fit $\frac{X}{2}$ bit integer range. The transformation uses a vector of features $\vec{f_o}$ that contains one individual feature from all samples in offline training data, then utilizes its mean and standard deviation (SD) for transformation (Eq. 5). The transformation applied to an original feature might result in values that are bigger or smaller than the $\frac{X}{2}$ bit space, which we cap at the boundaries (Eq. 6). This ensures that the $\frac{X}{2}$ bit space can be optimally used for the mainstream data, while boundaries are respected also for new, unseen data with potential outliers. The transformed vector of features $\vec{f_t}$ therefore consists of values in the range $[0, 2^{\frac{X}{2}} - 1]$, e.g. for 16 bit space the range of $[0, 255]$. This transformation is applied to all features.

$$\vec{f_r} = \text{round}\left(\frac{\vec{f_o} - \text{mean}(\vec{f_o})}{2 \cdot \text{SD}(\vec{f_o})}\right) \cdot (2^{\frac{X}{2}-1} - 1) + (2^{\frac{X}{2}-1} - 1) \tag{5}$$

$$\vec{f_t} = \begin{cases} 0 & \text{for } \vec{f_r} < 0 \\ 2^{\frac{X}{2}} - 1 & \text{for } \vec{f_r} > 2^{\frac{X}{2}} - 1 \\ \vec{f_r} & \text{else} \end{cases} \tag{6}$$

On mobile devices, the same feature preprocessing and simplification transformation is applied to features of new recordings during enrollment and authentication. Therefore, the mean and SD per feature computed from offline training data are stored on mobile devices outside the SC[5]. After simplifying features, the obtained simplified biometrics feature vectors are handed to the SC for purpose of enrollment or authentication.

5.3.1.2 *Model Training*

Offline model training uses pairs of samples represented by their feature vectors. At first, the distance between two biometric feature vectors $\vec{v_1}$ and $\vec{v_2}$ yields an absolute distance vector $d(\vec{v_1}, \vec{v_2})$ of same length, also in $\frac{X}{2}$ bit representation (Eq. 7).

$$d(\vec{v_1}, \vec{v_2}) = |\vec{v_1} - \vec{v_2}| \tag{7}$$

We refer to feature distance vectors originated by the same person as being of the positive class P and to those originated by different people as being of the negative class N. Using feature distance vectors

[5] Due to subsequent feature selection only a subset of those features remain. Storing and performing the simplification is only done for actually used features.

from our offline training data we create a classification model able to distinguish between the P and N class (for details on how data partitioning is done for model training and evaluation see Sec. 5.4). The obtained model can then be used on the mobile device to decide if a new feature distance vector is a P or N sample.

As classification model we use a linear discriminant analysis (LDA) model [139]. In contrast to the previously utilized [103] generalized linear model (GLM) [91], LDA aims to maximize the P-N inter-class-distance and minimize the P and N intra-class-distances of samples. Therefore, LDA models can usually provide for better class separation over GLM models. However, as both models are linear models, in their ready trained state both can internally be represented by a slope $\vec{s_o}$ (model coefficients) and an additional intercept I (offset to the origin of the coordinate system). For a distance vector $\vec{d}$ from a template and a new recording, those are used to predict the class membership C_d using a linear combination (Eq. 8, $\odot$ depicts the piece-wise multiplication of vector elements).

$$C_d = \begin{cases} P & \text{for } \sum_i \vec{s_o} \odot \vec{d} < I \\ N & \text{else} \end{cases} \tag{8}$$

Such linear combinations are simple enough to be computed on a SC, which is a core reason for choosing this model type. From training we obtain the optimal slope and intercept – which are later used to predict the class of new samples in both an offline evaluation of our generic MOC approach as well as the application case of on-device authentication.

5.3.1.3 *Model Simplification*

The slope $\vec{s_o}$ and intercept I obtained from model training are real-valued and, similar to biometric features, have to be simplified to enable their usage on a X bit integer SC. We therefore scale original model coefficients $\vec{s_o}$ to optimally fit a $\frac{X}{2}$ bit space and apply a cap at boundaries, resulting in a transformed slope $\vec{s_t}$ (Eq. 9 and 10). In contrast to transforming biometric features (Eq. 5), no shift is applied. This would otherwise change the meaning of coefficients, as coefficients around 0 have less influence on the result than those with higher absolute values.

$$\vec{s_r} = \text{round}\left(\frac{\vec{s_o}}{2 \cdot SD(\vec{s_o})}\right) \cdot (2^{\frac{X}{2}-1} - 1) \tag{9}$$

$$\vec{s_t} = \begin{cases} -(2^{\frac{X}{2}-1} - 1) & \text{for } \vec{s_r} < -(2^{\frac{X}{2}-1} - 1) \\ +(2^{\frac{X}{2}-1} - 1) & \text{for } \vec{s_r} > +(2^{\frac{X}{2}-1} - 1) \\ \vec{s_r} & \text{else} \end{cases} \tag{10}$$

Having both feature distance vectors and the slope in $\frac{X}{2}$ bit integer representation now allows for their piecewise multiplication on SCs in X bit integer range (Sec. 5.3.2). Therefore, this can be done efficiently on SCs that only support calculations in X bit integer range in hardware.

5.3.1.4 *Feature Selection*

After model training, features that are associated to small coefficients necessarily have small influence on the output – hence both feature and coefficient can possibly be removed without severely influencing classification performance. As selection criteria we thereby use the strongest absolute coefficient c_{max} as reference: a coefficient c_i is selected if it fulfills $c_i \geqslant \alpha \cdot c_{max}$, with α in the range $[0, 1]$. For details on used thresholds α and number of selected features for individual biometrics see Sec. 5.4.

By performing feature selection we achieve reduced storage requirements and computations on the SC, as well as reduced features to transfer to the SC, which therefore reduces the overall SC processing duration. Another, smaller advantage is that relying on stronger features could slightly increase overall predictive power of the model. However, as small coefficients do not necessarily denote features completely unimportant for separating classes, doing this might as well slightly reduce prediction capabilities.

5.3.2 *Mobile Device: Enrollment and Authentication*

Preparation of mobile devices comprises storing the feature normalization and simplification parameters on the mobile device, as well as storing the model (slope and intercept) directly on the SC. After data recording, enrollment and authentication perform data preprocessing, feature derivation, and feature simplification as stated in Sec. 5.3.1. On mobile devices those can be done outside the SC, as they do not use any information about templates previously stored on the SC. For enrollment, m feature vectors – derived from m newly recorded biometric samples – are transferred to the SC, where they are stored in the enrolled template for later usage. No further calculations are done on the SC. For authentication, n feature vectors from n newly recorded biometric samples are transferred to the SC. As this latter transmission is done for each authentication attempt, the transfer period is important and measured in our evaluation in Sec. 5.4.

On the SC we perform $m \cdot n$ comparisons between all m stored reference samples and all n newly transmitted samples using the stored, offline-computed model. To keep those $m \cdot n$ linear combination within a range of X bit (especially during summing intermediate, piecewise products of slope and difference vector), we utilize the mean value instead of a sum. Hence each intermediate product is immediately divided by the length of the slope vector to predict the class C_d (Eq. 11).

$$C_d = \begin{cases} P & \text{for } \sum_i \left(\frac{\vec{s_{t,i}} \cdot \vec{d_i}}{\text{length}(\vec{s_t})} \right) < I \\ N & \text{else} \end{cases} \tag{11}$$

The resulting $m \cdot n$ predictions, each indicating P or N class, are treated as votes. Using majority voting we compute a final, binary authentication decision from them, which is handed from the SC to the mobile device to authorize or deny an authentication attempt. If we would instead hand an authentication probability from the SC to the mobile device, this would conceptually allow for more flexible feedback to users. The downside of doing so is the danger of enabling hill climbing attacks to unlock the system or deriving information about users' biometrics (cf. [117, 222, 347, 355]), which is why we yield only binary authentication decisions from the SC.

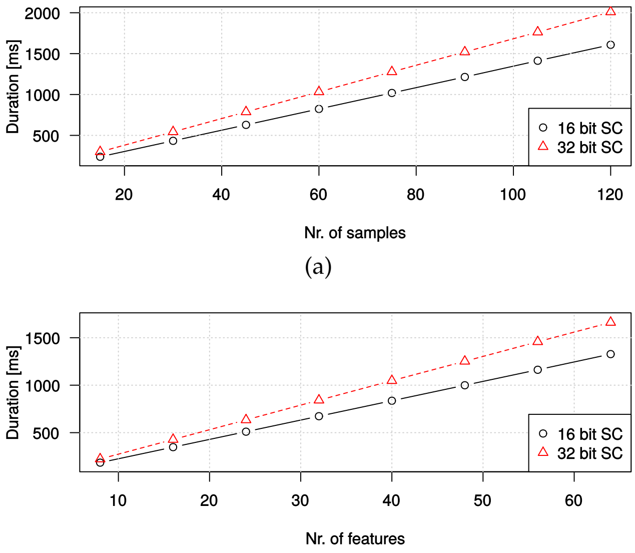
Besides allowing for linear combination in hardware on X bit SCs, our generic MOC approach has the advantage of requiring only $(n+2) \cdot \frac{X}{2}$ bits of storage memory on a SC for the model, when using n features (n corresponds to the slope, 2 corresponds to the intercept). For example, with 16 bit SCs, a model for 10 features could be expressed in only 12 bytes of SC storage. Similarly, m samples in an enrollment template require only $m \cdot n \cdot \frac{X}{2}$ bits of storage. For example, with 16 bit SCs, 8 samples consisting of 75 features require only 600 byte of SC storage.

5.4 EVALUATION

We evaluate our generic MOC approach on 16 and 32 bit SCs with face and gait biometrics, measuring both SC computation duration and authentication performance. We use a 16 bit JCOP 2.4.1 SC with 80 kB EEPROM memory running Java Card version 2.2.2 and a 32 bit SIM-card with 1 MB non-volatile memory and Java Card version 3.0.1. Communication was done over the contact interfaces of these cards using the same card reader.

5.4.1 *Duration on Smart Cards*

The duration of transferring one sample with 75 features to the SC and yielding an authentication decision back was measured to be on average 31.5 ms (SD=0.14 ms) with 16 bit SCs and 16.7 ms (SD=0.08 ms) with 32 bit SCs. This duration excludes computations on the SC and scales linearly with the amount of samples sent. Computing our complete approach on SCs also shows a nearly linear increase of computation time over both number of samples in the enrolled template and number of features per sample (Fig. 14). Those calculations include the computation of distances between samples in the enrolled template stored on the SC with newly transmitted samples, the linear combination of distances with model parameters determined offline, the voting of individual results to obtain an authentication decision, and the yielding thereof.



(a)

(b)

Figure 14: Average duration of our generic MOC approach on 16 and 32 bit SCs, including transmissions, for (a) different number of samples in the enrolled template, using 75 features per sample, and (b) different number of features per sample, using 32 samples in the enrolled template.

In absolute numbers, data transmission time becomes negligible compared to computation time on the SC. This implies that changing the number of samples m in the enrolled template and number of samples n in the new recording has little impact if the number of total votes $m \cdot n$ is unaffected. With using $m \cdot n = 64$ we achieve an average computation time of 1608 ms and 2010 ms for 16 and 32 bit SCs, and 824 ms and 1032 ms when using $m \cdot n = 32$ instead. The increased duration for 32 bit SCs has two reasons: a) twice the amount of data needs to be transmitted due to samples containing twice the amount of information as compared to 16 bit SCs. b) the amount of data that can be sent in one query is limited to 255 bytes by the transmission protocol of the SC (cf. application protocol data units (APDU) in [162]). Consequently, one 16 bit feature is transferred as two separate bytes, of which conversion to one 16 bit short on the SC requires additional time. While this limitation could be overcome by using the extended version of the protocol (extended length fields in [162]), in our measurements we consider the short and therefore slower variant for interoperability with all currently deployed smarts cards.

5.4.2 *Evaluation Setup for Using Different Biometrics*

To obtain realistic authentication performance estimates of people unseen by the model during training, we perform a non-overlapping, 50%/50% population independent split [164] on the corresponding datasets. We thereby assign 50% of participants to the training partition, which is used for training the model, and 50% of participants to the test partition, which is only used once for estimating the performance of the chosen and trained final model on yet unseen people. We further use only training data to determine parameters for feature derivation, simplification, and selection, then use the determined parameters to transform test data the same way. Within both training and test partition we use all combinations of different samples originated by the same person to obtain P distances and all combinations of samples originated by different people (within the corresponding partition) to obtain N distances.

The training partition is used to train and evaluate different parametrizations of our model to find a suitable configuration for distinguishing between P and N distances. As training and evaluation procedure we thereby use well established 10-fold cross validation with 10 repetitions and report the fit as receiver operating characteristics (ROC) curve, area under the ROC curve (AUC), and equal error rate (EER). After an optimal parametrization has been found (i.e. minimal coefficient threshold α and nr. of votes $m \cdot n$), the model is trained again using this configuration and all training data. The resulting model is evaluated once on the test partition to obtain a realistic authentication performance estimate on data of yet unseen people. For this we report the resulting true positive rate (TPR) and true negative rate (TNR). For comparability we additionally also report the ROC curve, AUC, and EER, when using all parametrization determined from training on the test partition, except the final decision threshold.

The resulting model further serves as basis for voting when using multiple biometric samples in both template stored on the SC and new recordings for authentication. Thereby, m cycles are contained in the enrolled template and n new recordings are provided during authentication – which results in a total of $m \cdot n$ samples and votes. For tuning the voting approach we use the same data partitions, with the training partition being used to evaluate the authentication performance of different amount of votes. Then, test data is again used only once for estimating the authentication performance for the final, voting based authentication model on data of yet unseen people.

5.4.3 *Evaluation with Gait Biometrics*

For evaluating our MOC approach with gait biometrics we utilize cycle based gait authentication based on acceleration data recorded by off-the-shelf mobile devices. In contrast to previous research on gait authentication we use a MOC approach, a non-DTW based model, and combine features previously used in acceleration gait recognition with features from other domains.

5.4.3.1 *Gait Data Source*

For our evaluation we utilize the acceleration gait database of Muaaz and Mayrhofer [242] which contains 3D acceleration recordings of 35 people, each walking about 550 m in total. The data was recorded with off-the-shelf smartphones featuring 100 Hz 3D accelerometers, with phones being placed realistically in trousers pockets. Further, for each participant, recording was split into two sessions with a gap of on average 25 days between recording, which allows for realistic cross-day evaluations of gait authentication systems. From this data we utilize cross-day, left-pocket recordings of all participants to train and evaluate our generic MOC approach with gait biometrics.

5.4.3.2 *Gait Data Preprocessing and Feature Derivation*

Preprocessing mechanisms are adapted from Nickel [254] as well as Muaaz and Mayrhofer [240, 242], which comprise of walking detection and preprocessing, as well as subsequent step detection and preprocessing, which we briefly summarize here. From 3D acceleration recordings, we extract walking segments with y-axis acceleration variance above $0.8\frac{m}{s^2}$ for at least 10 s. To compensate for gravity, we remove the mean acceleration segment and axis, then compute the resulting acceleration magnitude. As acceleration sampling is not necessarily uniform, we further perform a linear interpolation to obtain a uniform sampling rate of 100 Hz. For noise reduction we apply a Savitzky-Golay filter [301] with window length of 150 ms and polynomial of 1st order. The core advantage of this filter over frequently used running mean or median filters is the better retaining of the original signal shape.

For step cycle segmentation, reference cycles are extracted from each walking segment, around the middle of the segment [242]. Those are used to determine previous and successive starts of cycles in the same walking segment, which in turn are segmented into individual gait cycle samples of the corresponding individual. Furthermore, those are linearly interpolated to a uniform length of 100 acceleration values each, which correspond to a duration of 1 s at a 100 Hz sampling rate. Cycles that diverge largely from the majority of extracted cycles are further defined as outliers and discarded. For that purpose we compute the normalized dynamic time warping (DTW) distance[6] between all n cycles and discard those cycles for which more than $\frac{n}{2}$ distances are above a predefined threshold of 0.6. The remaining gait cycles are used in feature derivation and subsequently handed to the SC for enrollment or authentication (Fig. 15).

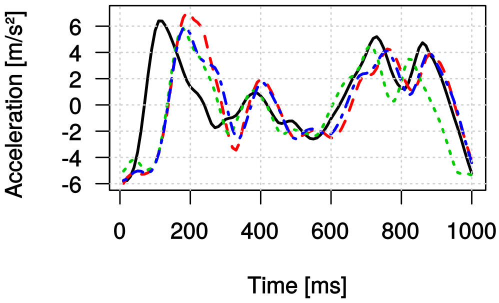


Figure 15: Examples of preprocessed gait cycles with a uniform length of 1 s, consisting of 100 values each.

For each preprocessed cycle we derive a number of features. In the time domain we utilize the mean, median, SD, median absolute deviation (MAD), and autocorrelation (AC) series with a maximum shift of 100 values as features on one cycle. AC has been used as signal preprocessing in other biometric recognition tasks, such as electrocardiography (ECG) recognition [25], but to our knowledge not yet in acceleration based gait authentication. To reduce naturally existing inter-feature correlation of the resulting AC feature vector, we use only every third value as feature. With a sampling rate of 100 Hz this corresponds to a shift granularity of 30 ms. In the frequency domain we compute the fast Fourier transformation (FFT) of the cycle. As human body motion sensed by accelerometers usually yield usable information in the frequency range of about 0-20 Hz (cf. [40, 107, 368]), we use both frequency power and phase in this range as features. Frequency power and phase are added as separate features to a) avoid passing complex values to models and b) enable separately treating them (e.g. normalizing and discarding features individually). Additionally, we also compute a discrete wavelet transform (DWT) representation of a cycle using a multiresolution analysis of 6 levels. As wavelet we utilize a least asymmetric Daubechies wavelet [78] of length 8. As with FFT features, all wavelet features are treated as individual features too. In total we thereby obtain a feature vector of length 177, which we can reduce to 64 features for both 16 and 32 bit SCs using a feature selection coefficient threshold of $\alpha = 0.35$. Therefore, with gait data our MOC approach requires 66/132 bytes of storage (for 16/32 bit SCs) for the offline computed model and 64/128 bytes per gait cycle in the enrolled template. With 8 cycles in the tem-

[6] This DTW distance calculation is done for data cleaning purposes outside the SC, consequently is not related to the authentication model and matching procedure on the SC.

plate this leads to a total of 578/1156 bytes of storage requirement on the SC.

5.4.3.3 *Gait Model Training and Authentication Results*

Due to slightly different amounts of gait cycles being discarded per participant during preprocessing and data cleaning, preprocessing results in a total of 2132 and 1943 unique gait cycles in the training and test partition, respectively. Due to the size of the training partition and the resulting training complexity, we use a random subset of 100000 P and 150000 N distances for training the model. However, for intra-training evaluation of trained models, the full training partition size is utilized (Tab. 2).

Partition	Cycles	P	N
Training	2 132	174 410	2 207 243
Test, pop. independent	1 943	168 976	2 158 427

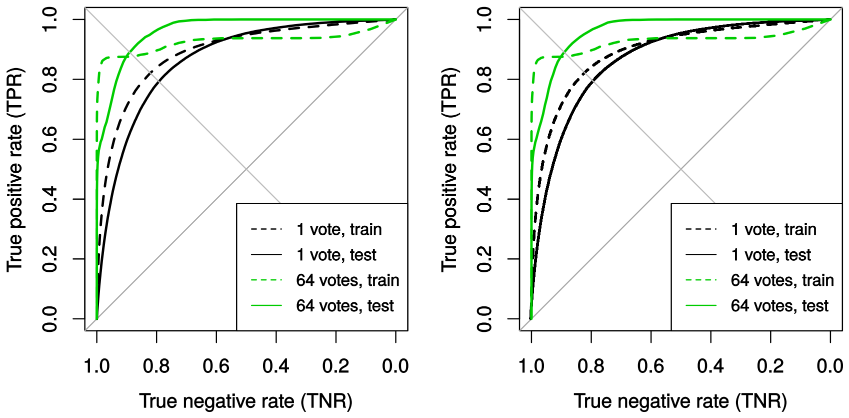
Table 2: Gait biometrics: training and test partition sizes, as amount of gait cycles and the resulting amount of P and N comparisons.

Gait evaluation results indicate a test partition EER of about 0.21 when using a single gait cycle in both enrolled template and new recording for authentication (Tab. 3 and Fig. 16). When using 64 comparisons instead (e.g. 8 samples in both enrolled template and new recording), we achieve an EER of about 0.114. With both, results differ only marginally between 16 and 32 bit SCs.

Partition	Votes	SC	AUC	EER	TPR	TNR
Training	1	16 bit	0.892	0.179	–	–
Training	1	32 bit	0.892	0.179	–	–
Test	1	16 bit	0.868	0.210	0.787	0.780
Test	1	32 bit	0.867	0.207	0.787	0.797
Training	64	16 bit	0.927	0.123	–	–
Training	64	32 bit	0.928	0.123	–	–
Test	64	16 bit	0.963	0.114	0.958	0.809
Test	64	32 bit	0.963	0.114	0.959	0.810

Table 3: Gait evaluation results for using a single gait cycle in both the template and the new recording and a total of 64 votes (e.g. 8 templates and 8 new recordings to compare to).

These results indicate that for acceleration based gait data, increasing the granularity of model coefficient and feature space – as required for usage of our MOC approach on 16 bit SCs – does not lead to considerably worse results over using 32 bit SCs, where the resolution of data is allowed to be twice as fine. Using the feature space available with 16 bit features and model coefficients on 32 bit SCs results in longer durations, caused by higher feature precision and the



(a) 16 bit SC

(b) 32 bit SC

Figure 16: ROC curves for using a single gait cycle in both the template and the new recording and a total of 64 votes (e.g. 8 templates and 8 new recordings to compare to).

corresponding higher total amount of data transferred and processed. Further, our results with using SCs also seem comparable with findings from previous research without SCs on the same dataset with 18% EER when comparing single gait cycles [241] and 94% TNR and 64% TPR when using 4 gait cycles in one comparison [242]. In contrast to our approach those approaches rely on a computationally intensive DTW unsuitable for computation on SCs. In comparison to the latter result, our approach shows an improved TPR and worse TNR –which corresponds to lower obtrusiveness, but also lower security. To achieve a higher and thereby comparable TNR with our approach two options would be possible: a) adapting the decision threshold, hence choosing a different point in the corresponding ROC curve to achieve a higher TNR at the cost of a lower TPR. This would cause security to be increased (less likely for attacks to be successful), but also cause the approach to be more obtrusive (more frequently rejecting legitimate users). b) using more comparisons of gait cycles to derive an authentication decision. This would lead to an increased TNR and TPR at the cost of longer delays caused by increased calculation durations and/or longer walking time until authentication is performed.

5.4.4 *Evaluation with Face Biometrics*

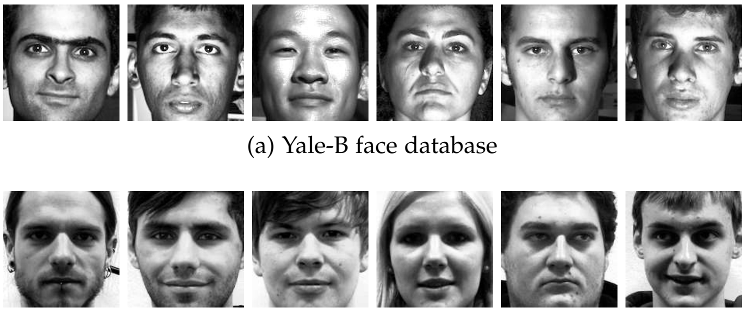
For demonstrating our MOC approach with face biometrics, we use view-based face authentication based on 2D wavelet transformed representations of face images and estimate the authentication performance with two publicly available face databases.

5.4.4.1 *Face Data Source*

To demonstrate our MOC approach on face biometrics we use subsets of the Yale-B [194] and the Panshot Face Unlock Database [102]. The Yale-B database contains facial images illuminated with a light source from different azimuths and elevations relative to the face. We thereby utilize face images with maximum azimuth and elevation of ±20° between light source and face, which results in a database subset 511 facial images of 27 participants. In contrast, the Panshot Face Unlock database contains face images recorded from 9 different perspectives in a 180° semi circle around the head using different recording hardware. We thereby utilize facial images recorded from a frontal perspective, which results in a total of 600 images of 30 different participants. For both databases, we use grayscale, unsegmented (neither face-detected nor cropped) images, then perform face detection and segmentation ourselves to obtain faces realistic for a mobile authentication scenario.

5.4.4.2 *Face Data Preprocessing and Feature Derivation*

At first we equalize the image histogram per image, then perform Viola and Jones face detection [356] to detect and segment the part of the image related to facial information into quadratic images. We only consider the face image if its diagonal is at least $\frac{1}{4}$ the diagonal of the original image. In mobile face authentication scenarios, where users are within arms reach of their mobile device, requiring such a relative minimal face image size effectively prevents a large portion of potential false positive face detections. Further, if multiple faces are detected, we only consider the biggest detection. We again equalize the histogram per face image. Equalization results are different than before face segmentation, as background information that contributed to the equalization has now been removed from the images (Fig. 17).



(a) Yale-B face database

(b) Panshot Face Unlock database

Figure 17: Examples of preprocessed, segmented, and equalized face images from the Yale-B and Panshot Face Unlock databases handed to feature derivation [102, 194].

Before deriving features, we downscale images to reduce processing power required in subsequent steps on mobile devices and SCs. In preliminary experiments we used face image sizes of 64×64 and 32×32, in which the latter turned out to be sufficient for subsequent feature derivation and MOC face authentication. We therefore used face images of size 32×32 – but our MOC approach could be applied analogously to other image sizes as well. As feature derivation we use 2D discrete wavelet transformation (2D-DWT) and multiresolution analysis with a Daubechies Least-Asymmetric 2D Wavelet [78]. The resulting coefficients are treated as feature vector of length 1365, which can be reduced to 75 features (16 bit SC), respectively 72 features (32 bit SC), using a maximum feature coefficient threshold $\alpha = 0.95$. Therefore, with face biometrics our MOC approach requires 77/148 bytes for storing the model (with 16/32 bit SCs) and 75/144 bytes per face in the enrolled template. With 8 face images in the template this leads to a total storage requirement of 677/1300 bytes.

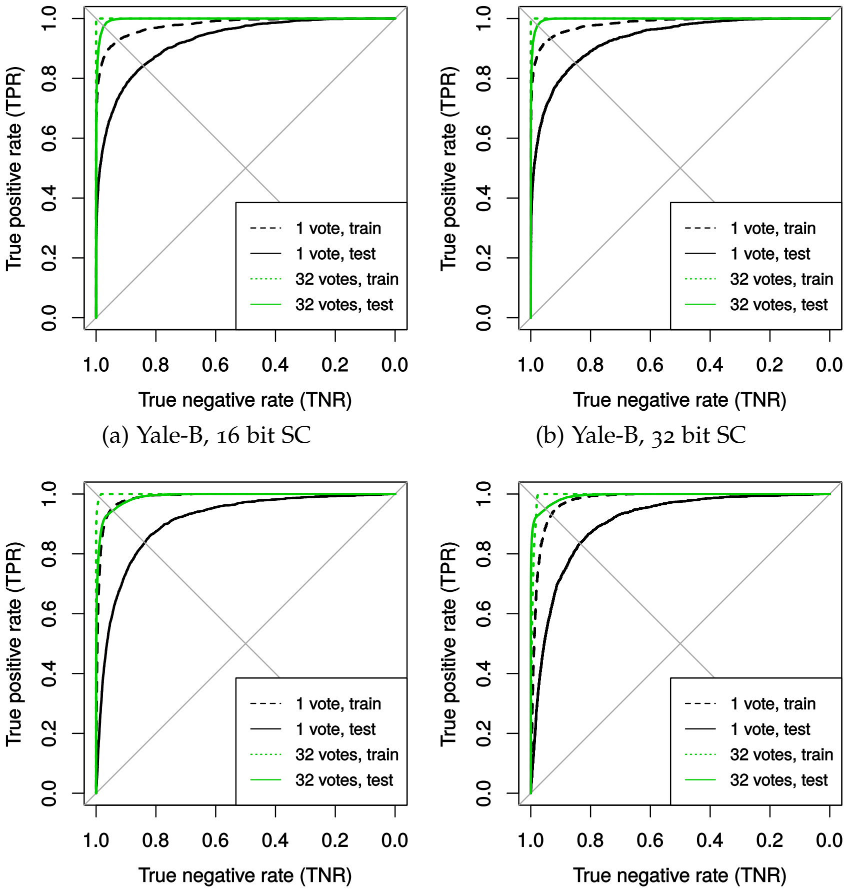
5.4.4.3 *Face Model Training and Authentication Results*

Due to slightly different amounts of faces detected per participant we obtain slightly different training and test partitions for both databases (Tab. 4).

Database	Partition	Faces	P	N
Yale-B	Training	265	2 376	32 604
Yale-B	Test	246	2 205	27 930
Panshot	Training	296	2 780	40 880
Panshot	Test	273	2 536	34 592

Table 4: Face biometrics: training and test partition sizes, as amount of face images and the resulting amount of P and N comparisons.

Similar to the results of the gait based evaluation, authentication performance differs only slightly between 16 and 32 bit SCs (Tab. 5 and Fig. 18). Using the Yale-B database we obtain a test partition EER between 15-16% without majority voting of comparisons of multiple face images. Additionally employing a majority vote boosts results to 2.4-3% EER. Using a 32 instead of 16 bit SC marginally increases the overall authentication performance, visible in both decreased EER and increased AUC. Using the Panshot Face Unlock database, we obtain a slightly worse test partition performance of 16.3% EER without majority voting, which is decreased to 5.3-5.4% EER using majority voting. We assume that results being worse is due to the Panshot Face Unlock database containing faces with less distinctive features recorded more uniformly, which makes distinguishing them more difficult. Overall, results confirm that our generic MOC approach is also applicable to both types of SCs with facial biometrics. Similar to gait results, the gain of using a 32 instead of 16 bit SC is minimal with face



(a) Yale-B, 16 bit SC

(b) Yale-B, 32 bit SC

(c) Panshot, 16 bit SC

(d) Panshot, 32 bit SC

Figure 18: ROC curves for using a single face image in both the template and the new recording and a total of 32 votes (e.g. 8 templates and 4 new recordings to compare to) for training and test partitions.

Database	Partition	Votes	SC	AUC	EER	TPR	TNR
Yale-B	Training	1	16 bit	0.980	0.075	–	–
Yale-B	Training	1	32 bit	0.983	0.067	–	–
Yale-B	Test	1	16 bit	0.925	0.159	0.890	0.775
Yale-B	Test	1	32 bit	0.932	0.150	0.900	0.784
Yale-B	Training	32	16 bit	1.000	1.000	–	–
Yale-B	Training	32	32 bit	1.000	1.000	–	–
Yale-B	Test	32	16 bit	0.997	0.030	0.998	0.933
Yale-B	Test	32	32 bit	0.998	0.024	0.996	0.954
Panshot	Training	1	16 bit	0.987	0.051	–	–
Panshot	Training	1	32 bit	0.977	0.070	–	–
Panshot	Test	1	16 bit	0.909	0.163	0.754	0.892
Panshot	Test	1	32 bit	0.907	0.164	0.748	0.885
Panshot	Training	32	16 bit	0.999	0.012	–	–
Panshot	Training	32	32 bit	0.995	0.022	–	–
Panshot	Test	32	16 bit	0.990	0.054	0.792	0.992
Panshot	Test	32	32 bit	0.993	0.053	0.797	0.999

Table 5: Face evaluation results for using a single face image in both the template and the new recording and a total of 32 votes (e.g. 8 templates and 4 new recording to compare to) for training and test partitions.

biometrics. Therefore, using the increased resolution of feature space and model coefficients available with 32 bit SCs seems unnecessary, as it primarily leads to an increased duration of our MOC approach due to bigger amount of data transferred and processed.

5.5 SUMMARY

For mobile biometric user-to-device authentication we proposed to train match-on-card (MOC) authentication models offline using machine learning. We use model types that feature a simple internal representation once they are fully trained. To enable their usage on SCs, we adapt and simplify both used features and models. The model is computed only once using a dataset of the corresponding biometrics, then stored on SCs of mobile devices. Enrollment on mobile devices involves recording samples of the authorized user and storing their feature vectors on SCs without requiring retraining the model. Authentication compares features of newly recorded samples with enrolled samples on the SC, using the previously stored model to derive a binary authentication decision. One major advantage of the proposed approach is that it is generic and can be applied on different biometrics alike, thereby facilitating the translation of mobile biometric matching procedures towards MOC in general.

We applied our generic MOC authentication approach to acceleration based mobile gait authentication as well as face authentication, utilizing both 16 and 32 bit Java Card SCs. With gait authentication, when using 8 cycles in the enrolled template and 8 newly recorded cycles for authentication, we found our approach to be feasible with an EER of 11.4%. Authentication time on the SC stays in the range of 2 s, including data transmissions and authentication computation. To the best of our knowledge this work represents the first practical approach towards acceleration based gait MOC authentication. With face authentication, when using 8 face images in the enrolled template and 4 newly recorded face images for authentication, we found our approach to be feasible with an EER of 2.4-5.4% EER. The authentication time on the SC thereby stays in the range of 1 s, again including both transmission and calculation time on SCs. We argue the durations of 2 respectively 1 s to be a reasonable trade-off between authentication performance and delay, as responsiveness will usually be more critical for face than gait authentication. This is because face authentication can be performed actively, where users expect immediate authentication results – while gait authentication is done as passive, unobtrusive background authentication, therefore is less sensitive to higher authentication latency. Using 16 instead of 32 bit SCs seems to have little negative impact on authentication performance. From this we derive that an adequate representation of samples and models is possible in the more granular feature and model coefficient space on 16 bit SCs. Furthermore, using the higher resolution of information of 32 bit SCs leads to more data being transferred and more computations on SCs, which overall make the approach slower than on 16 bit SCs.

To summarize, these results indicate that our generic mobile MOC authentication approach is feasible and can be applied to different biometrics on both 16 and 32 bit SCs. In the future, it might thereby facilitate the transfer further mobile biometrics toward using MOC techniques. This would further aid mobile authentication being unobtrusive in different situations (using different biometrics suiting those situations) – without exposing their user to the additional risk of disclosing their biometrics.

6 TRANSFERRING AUTHENTICATION STATES BETWEEN DEVICES BY SHAKING THEM CONJOINTLY

In this chapter we highlight our token-based user-to-device authentication approach which utilizes brief conjoint shaking of mobile devices to transfer the authentication states between them (Fig. 10, b). Parts of this chapter have previously been published in [106, 107].

Token-based authentication approaches in the mobile environment that purely use proximity to derive an authentication decision have the drawback of attackers possibly being able to unlock mobile devices they got under their control just by being close to the user. For example, with tokens relying on a distance derived from WiFi or Bluetooth signal strengths it might be sufficient for attackers to be in the same room with the legitimate, inattentive user to successfully unlock the device. As attackers are likely to be close to the user when obtaining control over the mobile device, an immediate unlock would be possible before leaving the scene. When using token-based authentication, the token needs to be brought by users everywhere they potentially want to use their mobile device. Depending on where the token is kept, it could be possible to obtain control over both token and device at once and then use the token to unlock the device. If the token itself is locked to prevent illegitimate usage in case of theft, the whole problem is transfered from the mobile device to the token – as unlocking the token itself again could be done using knowledge-, biometrics- or token-based authentication.

To address these issues we propose a novel token-based mobile device unlocking approach: transferring the authentication state between two devices by briefly shaking them conjointly. The key idea is that personal mobile devices can remain unlocked for different periods of time, one could act as a token, allowing to transfer authentication state between devices. For example, a mobile phone should lock itself as soon as it is put aside while a smart watch could remain unlocked as long as it is strapped to the wrist and automatically lock itself when detached. The smart watch could e.g. be unlocked once in the morning when attached to the wrist and automatically lock itself when detached, utilizing e.g. heart-rate measurements like with the Apple Watch[1] or a simple connection in the strap that is triggered by opening it. Using this setup, the authentication state from the unlocked watch can be transferred to the locked phone to unlock it –

[1] Apple Watch heart rate measurements: https://support.apple.com/en-us/HT204666

hence the unlocked device can serve as token for unlocking other devices. Shaking both devices simultaneously with the same hand serves as a fast, easy and secure trigger for authentication state transfer. The authentication state transfer is only triggered after an analysis of sensor time series recorded on both devices concludes that a) both devices have been shaken simultaneously and b) both devices have been shaken by the same person. For simplicity, from now on we will refer to the device from which the authentication state is transferred as *token device* where applicable.

Unlocking mobile devices by shaking them conjointly has noteworthy advantages over other unlocking approaches. Required user attention is assumed to be lower compared to current unlocking approaches, as users only need one hand and are not required to look at the devices to unlock them. In terms of speed we aim for 2 s of shaking to transfer authentication states between devices to be comparable to other unlocking mechanisms (cf. studies showing that mobile unlocking duration ranges from 1.5 s for PIN entry to 3 s for graphical patterns [150, 377]). We assume that 1–3 s can be considered an acceptable unlocking delay for our scenario in terms of usability vs. security, while requiring less user explicit attention. Shaking devices can be utilized on a broad range of mobile devices nowadays as accelerometers are a common feature of mobile phones, tablets and smart watches as well as activity trackers and other wearable computing gadgets. Previous research on pairing mobile devices by shaking them conjointly has stated shaking to be secure, as acceleration records are difficult to forge by shaking devices bare handed [224], making it a suitable choice for security critical applications[2]. We base ShakeUnlock on these findings but focus on a different use case: transferring authentication states from a token device to another device to unlock it. Consequently, the scenario presented here implies different approaches towards security and usability with analyzing acceleration sensed on both devices. Our work focuses on the technical aspect and security implications of ShakeUnlock – and leaves a thorough evaluation of usability and acceptance for future work, as such a study would need to consider longitudinal effects of muscle memory/muscle learning (users being able to perform movements without explicitly thinking about them, like 10-finger-typing on a keyboard).

ShakeUnlock contributes to unobtrusive mobile authentication by providing an additional unobtrusive authentication option for different situations than addressed by existing approaches, that further does not impose cognitive load on users, and that allows for authentication with a duration in the range of 2 s. Summarizing, the contributions of ShakeUnlock are:

[2] Hypothetical attacks could involve e.g. high speed cameras and an apparatus to precisely recreate visually observed shaking behaviors but are beyond the scope of this work.

- In contrast to previous research on shaking mobile devices conjointly to establish a secure channel between them, we focus on shaking as a secure trigger mechanism to transfer authentication states from a token device to another device over a pre-established secure channel.
- ShakeUnlock processes data from mobile devices situated 10-15 cm apart from each other (mobile phone held in the hand, smart watch strapped to the wrist) with the wrist as a non-static joint in between, which implies differences in sensed acceleration on both devices.
- Using this setup we record the ShakeUnlock database containing 3D acceleration and 3D gyroscope time series recordings of mobile devices being shaken conjointly. We use this data to parameterize and evaluate ShakeUnlock.
- We give detailed insight into the time series similarity data analysis of ShakeUnlock. We evaluate the influence of shaking devices while sitting/standing or using the dominant/non-dominant hand, as well as the contribution of constituent parts to the overall system performance. We believe that future approaches can benefit from these detailed insights and findings.
- We implement ShakeUnlock on Android and present a performance study which evaluates three different attack scenarios.

6.1 SHAKING MOBILE DEVICES CONJOINTLY

6.1.1 *Previous Work on Analyzing Conjoint Movement of Mobile Devices*

Analyzing movement and acceleration records for determining if mobile devices were shaken together by the same body movement has been subject of a significant body of research over the last 10 years. Research ranges from analysis of simple movements with accelerometer recordings (cf. [13, 146]) to deriving secret keys from acceleration data (cf. [7, 30, 130, 180, 224, 310]).

With “Smart-Its Friends”, Holmquist et al. [155] have been amongst the first to associate devices by shaking them together. Their devices sense acceleration and broadcast it, so that other devices may decide on pairing with them. Their approach purely focuses on pairing without taking security aspects like Man-in-the-middle (MITM) or replay attacks into account. In “Are You with Me?”, Lester et al. [198] have built upon this work but use frequency domain based magnitude squared coherence instead of time domain based analysis to pair devices. Their approach has further been extended by Mayrhofer and Gellersen in “Shake Well Before Use” [224] which additionally covers security aspects of pairing devices by shaking them conjointly.

“Shake Them Up” by Catelluccia and Mutaf [56] utilizes a related idea, although it does not involve sensing acceleration. They monitor WiFi received signal strength indication (RSSI) which is likely to change when devices are moved/rotated. As devices are moved together they experience similar changes in RSSI over time on the basis of which devices decide if they have been moved together. This approach is designed with MITM protection in mind. However, it depends on wireless signals and wireless signal strength sensing capabilities to be available on both devices.

The special aspect of shaking devices conjointly which are apart from each other and have a non-static joint (e.g. the wrist) in between was addressed by Fujinami and Pirttikangas [112] for associating objects with users. Amongst other things they consider toothbrushing with sensors attached to the users hands and toothbrushes. Similarly, Bao and Intille [20] have investigated activity recognition including tooth brushing from 2D acceleration sensors and time domain features. We deal with the same complicating issues for robust acceleration time series comparison due to having a non-static joint between devices, which will cause devices to sense slightly different acceleration during shaking. Additionally, we have to consider security implications of attackers trying to forge acceleration patterns to get access to obtained devices.

In terms of data analysis, shared movement and shaking has been analyzed in both time and frequency domain. For in depth comparison we refer to [76, 77] as well as related research from the field of activity recognition (cf. [11, 96, 158]). Although analysis in time domain seems to be capable of yielding higher entropy [130], analysis in frequency domain seems more resistant to synchronization issues [198]. In ShakeUnlock, devices independently record acceleration and decide if they are currently shaken. Devices will sense slightly different acceleration due to the non-static joint in between them, hence detect active shaking at slightly different points in time. As we cannot assume exact synchronization between devices we use frequency based analysis. So far the most successful analysis approach is using frequency-domain based magnitude squared coherence [363], which has been used in various previous studies (cf. [28, 70, 131, 198, 220, 224]) and which is utilized in ShakeUnlock as well.

6.1.2 *Implications of Shaking on Security*

In 2011, Studer et al. [324] proved the well known and by now discontinued mobile phone application “Bump”[3] to be insecure. With “Bump” and similar approaches such as simultaneously pressing a button on both devices (cf. [146, 283, 318]) correct timing is the only critical aspect to establish a channel between devices. As timing can-

[3] See http://bu.mp

not be assumed secret, attackers can easily perform MITM attacks by forging required information and communicating them with correct timing. Instead of using timing constraints we utilize shaking to trigger the transfer of authentication state from the token device to other devices. Consequently, resistance against forged shaking patterns is required to prevent attackers from triggering an authentication state transfer without being in control of both devices at the same time.

Most previous research on shaking mobile devices conjointly in the scope of security aim to establish a secure channel between devices [30, 130, 224, 225, 227] (also known as bootstrapping or human verifiable authentication problem [67]). In contrast to these approaches we study shaking as trigger mechanism to transfer an authentication states from the token device to other devices over an pre-established secure channel.

6.2 THREAT MODEL

We want to emphasize that a) a user in control of the unlocked token device and the locked phone is intentionally able to trigger the authentication state transfer to unlock the phone, as no biometric authentication is performed. b) the authentication state transfer is triggered if – and only if – the token device is unlocked and the phone is locked when both devices are shaken conjointly, which renders being in control of the locked token device and phone insufficient for attacks. Consequently, access protection for the token device is required. As discussed before, when assuming that users attach their locked token device to their wrist once a day, then unlock it (e.g. in the morning), the token device can stay unlocked until users lock it manually or it is detached from the wrist. Compared to access to an unlocked phone or regular authentication token not featuring a locking mechanism, we argue that this brings an increased level of access protection to the unlocked token device:

- It is more difficult for the token device to be lost or stolen, as it is attached to the user's wrist.
- For attackers it is more difficult to obtain/access to the unlocked token device, as it automatically locks itself when detached from the wrist and accessing it in an unlocked state therefore would require accessing it before detaching it from users wrist, which is unlikely to go unnoticed.

For our scenario we therefore assume the token device to be secure and restrict addressed attack scenarios to the locked phone being under control of an attacker. We further assume that the token device is unlocked, as otherwise no authentication state transfer can be triggered.

6.2.1 *Attack Scenarios*

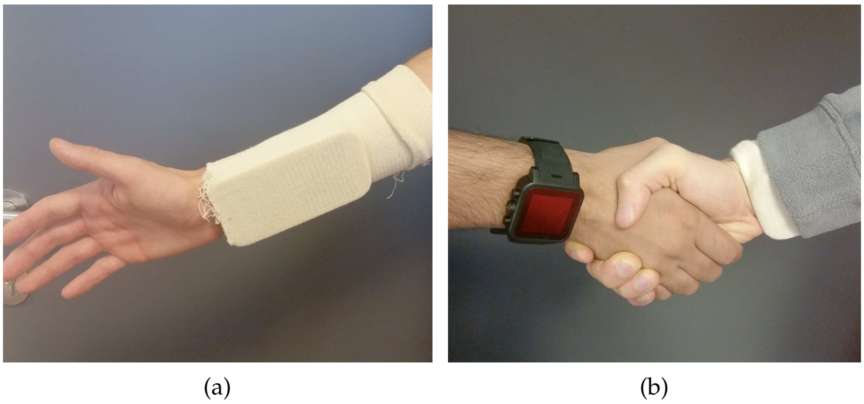
For all attack scenarios, the locked mobile phone is considered to be under physical control of an attacker trying to unlock it unnoticed by legitimate users who control the token device. To trigger an authentication state transfer from the unlocked token device to the phone, simultaneous shaking of both devices is required. This implies the legitimate user also has to shake the token device, which is why an attacker must synchronize any attack attempts with the user's shaking of the token device. We address four such attack scenarios with different attacker capabilities:

Minimal effort attacks assume that users have been tricked into accepting a proxy device as their own and subsequently try to unlock it by shaking it conjointly with the token device. Attackers simultaneously shake the target device they control but without trying to mimic the shaking pattern of users. Note that we use the term "minimal effort" because attackers do not take additional effort such as imitating users' shaking behavior. Sophisticated preparation, e.g. obtaining control over the device beforehand and tricking users into taking a different device for their own, is still required for this kind of attack. Being resistant against minimal effort attacks means being resistant against two people separately shaking both devices at the same time to trigger an authentication state transfer.

Observatory attacks use the same setup as minimal effort attacks, but attackers are observing the legitimate users and attempt to synchronously mimic the users' shaking patter to unlock the device, without the legitimate users noticing.

Cooperative attacks allow any cooperation between user and attacker except touching each other or the other's device in order to achieve high similarity in shaking patterns. This attack is supposed to break the approach and serve as measure of upper boundary to the security achieved, as in terms of authentication it is both unrealistic and harder than both previous attacks.

Handshake attacks assume attackers strap the mobile phone to their wrist using a bandage (Fig. 19). Then users and attackers shake hands hard to achieve synchronized acceleration records on both devices. This requires the hand to which wrist the token is attached to be used for the handshake. As with cooperative attacks, handshake attacks are supposed to break the approach. In a real life scenario, attackers shaking users' hands as hard as required to trigger recording of continuous 2 s shaking would be unrealistic, as it is far from natural and would make users suspicious.


(a)

(b)

Figure 19: Possible handshake attack setup with (a) the mobile phone being strapped to the attacker's wrist and (b) attacker shaking the user's hand hard.

6.2.2 *Attack Evaluation*

From security perspective, evaluating these attacks scenarios could be done with a one-to-one matching of data aggregated from devices both shaken and not shaken conjointly. These can be used to state a) success rates of legitimately triggering authentication state transfer (true positive rates) and b) attack success rates (false positive rates). From a system parametrization perspective, a larger number of samples is required to obtain suitable distinguishing capabilities. We therefore use m-to-n matching of uncorrelated shaking samples in our data set to simulate minimal effort attacks which we use in turn to parameterize ShakeUnlock (Sec. 6.5). To evaluate the remaining three attack scenarios we use an implementation of the proposed concept on off-the-shelf Android devices with one-to-one matching of live data (Sec. 6.6).

6.3 AUTHENTICATION STATE TRANSFER BY SHAKING DEVICES CONJOINTLY

ShakeUnlock is split into two major steps: separately sampling acceleration on both devices and deciding upon triggering an authentication transfer between devices on one device (Fig. 20). The first step consists of monitoring acceleration, deciding if the device is shaken, and extracting an active shaking acceleration segment (*active segment*) independently on both devices. If active segments have been detected, both are aggregated on one device. In the second step the similarity of active segments is determined to decide if devices have been shaken conjointly and thus an authentication state transfer should be triggered. Note that in contrast to related approaches, no acceleration

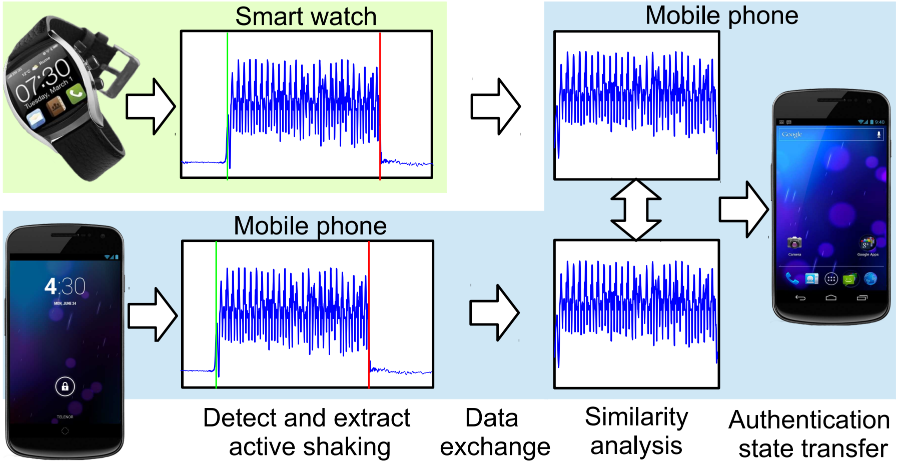


Figure 20: Data processing chain used in the ShakeUnlock approach.

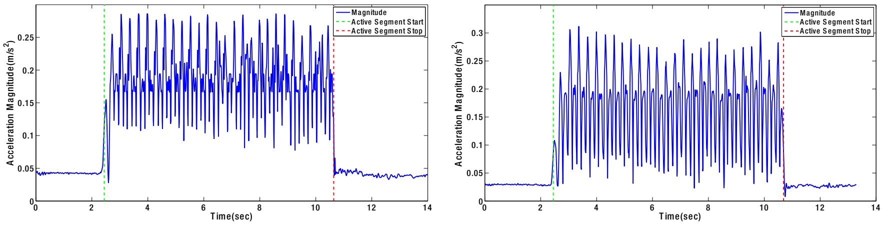
data is stored on the devices – not even in the form of cryptographic keys or hashes.

6.3.1 *Active Segment Detection*

In ShakeUnlock devices continuously and separately monitor acceleration, which can be done without excessive draining of battery power by utilizing hardware dedicated to acceleration recording. Such hardware is already becoming available in off-the-shelf mobile devices, such as for background step counting in the Apple iPhone5, iPhone6 and Apple Watch, Samsung Galaxy S5 or Sony Xperia Z1(c)-Z3(c) devices. As shaking is detected, the power efficient hardware can e.g. power on the main CPU which then performs the computationally more expensive networking and time series comparisons tasks.

ShakeUnlock determines the start of an active segment by monitoring the variance of the acceleration magnitude of the 3D acceleration sensor in a sliding window as described in [224]. If the variance of acceleration within this window rises above a certain threshold, this marks the start of an active segment from which acceleration on 3 axes is recorded for a short duration, capturing the shaking of the device. For our evaluation and implementation we use an acceleration monitoring sliding window of 2 s, an acceleration variance threshold of $6 \cdot 10^{-4} \frac{m}{s^2}$ and record active segments of 2 s length after shaking is detected. If users prematurely stop shaking (i.e. active segment < 2 s), no authentication state transfer will be triggered.

After active segments have been detected and recorded separately on both devices, we aggregate them on one device. Data aggregation could be done on each of the devices, as both are assumed secure and connected via a secured channel. However, when transferring the authentication state from the watch (token) to the phone, data aggregation on the phone has the following advantages: a) Usually, mobile



(a) Active segment detected on mobile phone

(b) Active segment detected on wrist watch

Figure 21: Active segments detected independently on the mobile phone and wrist watch.

phones have higher computational power than smart watches, hence the decision on performing the authentication state transfer will be obtained faster. b) If we conclude to perform the authentication state transfer from watch to phone based on recorded active segments, no further data transfer between devices is required, as the decision is done on the phone already.

6.3.2 *Authentication Transfer Decision*

After active segments have been recorded on both devices individually and aggregated on one device, we analyze those active segments to determine if devices have actually been shaken conjointly. If so, we perform an authentication state transfer between devices to unlock the device still locked. Before performing the actual similarity analysis, we preprocess the two active segments. We compensate for gravity recorded within the active segments by subtracting the mean acceleration per axis throughout the active segment.

Our similarity analysis takes a pair of active segments as input and yields a scalar metric value as output. If this metric value is above a reference threshold, we conclude that active segments represent devices shaken conjointly, therefore trigger the authentication state transfer and unlock the locked device. If the metric value is below the predefined threshold, we conclude that active segments represent devices not shaken conjointly, therefore refuse the authentication state transfer and do not unlock the device. Our similarity analysis consists of different constituent parts, which we present and discuss in Sec. 6.5.

6.4 EVALUATION DATA: THE SHAKEUNLOCK DATABASE

We recorded the *ShakeUnlock database*[4] which consists of 29 participants shaking a wrist watch (strapped to their wrist) and mobile phone (held in the hand). For each participant, we recorded 5 shaking samples each for four different setups (Tab. 6), which results in 20 samples per participant and device, and to 1160 samples in total – which overall reflect large differences in shaking style, vigor, and frequency.

Setup	Watch	Phone	Posture
1	left wrist	left hand	sitting
2	right wrist	right hand	sitting
3	left wrist	left hand	standing
4	right wrist	right hand	standing

Table 6: The u'smile ShakeUnlock database features 5 samples for each 4 different setups per participant.

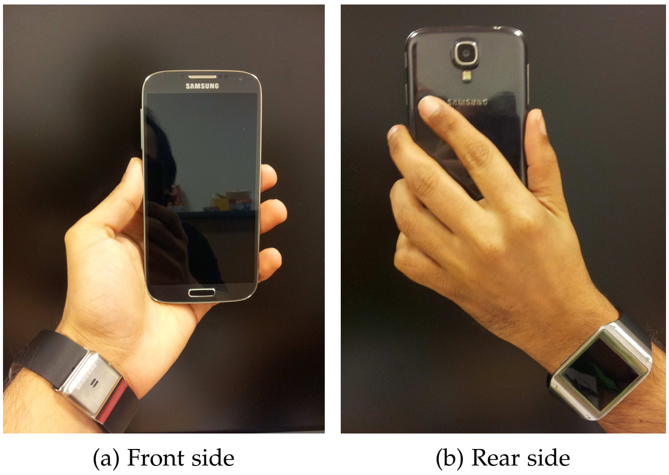
For data collection, we used an Android application recording 3 axes accelerometer time series and storing them in the form of comma separated value files locally on each device. The devices are connected over a Bluetooth channel, sending start/stop recording instructions as well as experiment metadata (e.g. subject ID) in a synchronized fashion when starting/stopping data recording. We explicitly note that this synchronization is only facilitating an easier experiment, but that it is not required for real-world use outside the recording setup.

Before data recording, participants strapped the watch to their wrist and grabbed the phone with the same hand (Fig. 22). Immediately before starting data recording, all participants were given the same, brief instructions: "Shake the devices as you would shake them intuitively, but shake them a bit harder/a bit quicker and try to not bend your wrist while shaking.".

Each recording has a total length of 13 s: 10 s of active shaking and 3 s of neutral device movement. Participants started the recording by pressing a button on the mobile phone and started shaking. They were informed to stop shaking by audio and vibration feedback from the phone after 10 s of recording (therefore active shaking is close to 10 s for most samples) – with the devices continuing to record for 3 s after the notification.

In total we recorded data from 25 male and 4 female participants, with an average age of 27 years and from different backgrounds and professions (we do not distinguish by profession, age or gender as it does not seem important for performing a simple shaking movement). Further we used a mix of different devices running Android

[4] The ShakeUnlock database is publicly available for download at http://usmile.at/downloads.


(a) Front side

(b) Rear side

Figure 22: Phone and watch placement for all setups, with the watch being strapped just as hard as necessary to prevent slipping during shaking.

Pair of devices	Male	Female	Total
Galaxy S4, Galaxy Gear	23	3	26
Moto G, Simvalley watch	2	1	3

Table 7: Amount of recordings done per pair of devices and gender of participants.

4.0 or above (Tab. 7). For 26 participants we used a Samsung Galaxy S4 mobile phone (model GT-I9500) together with a Samsung Galaxy Gear wrist watch (model SMV700). For the remaining 3 participants we used a Moto G mobile phone (XT1032) together with a Simvalley Mobile wrist watch (model AW-420.RX) to analyze how dependent various parameters of the data analysis pipeline are on the specific recording hardware. The recording acceleration sensor sampling rate was fixed on operating system side to 100 Hz. Therefore, any inaccuracies in sample timing are caused by the operating system itself and would also occur in implementations of ShakeUnlock on other platforms.

6.5 ACTIVE SEGMENT SIMILARITY ANALYSIS

Previously Mayrhofer and Gellersen [224] showed that it is feasible to detect if devices – which are pressed against each other – have been shaken conjointly using magnitude squared coherence on acceleration time series magnitudes. We adapt this method in order to apply it to acceleration time series magnitudes of devices somewhat apart and with non-static joint in between during shaking. Our approach thereby incorporates different preprocessing and parametrization. It further incorporates what we refer to at constituent parts of ShakeUnlock: additional derotation of 3D time series before performing the similarity analysis, bandpass filtering, a different collapsing function, and optimal weighting of individual frequencies. In this section we at first evaluate the impact of shaking devices for different durations as well as shaking devices when standing/sitting or using the dominant/non-dominant hand. We further evaluate the influence of each constituent part of our proposed approach on the overall performance. Thereby obtained performance comparisons are stated in Sec. 6.5.9.

6.5.1 *Parametrization and Evaluation Data Partitioning*

We parametrize and evaluate ShakeUnlock using acceleration data from the ShakeUnlock database on the basis of a) devices being shaken conjointly and b) simulated minimal effort attacks. Other attack scenarios are not based on this but use separately recorded data (Sec. 6.6). We at first extract active segments (Sec. 6.3) for all these samples which simulate users shaking their devices to transfer the authentication state. We then use active segments from all 580 time series pairs of devices shaken conjointly as legitimate tries to trigger authentication state transfer between devices. Therefore, our positive class P is of size 580. To simulate minimal effort attacks we use all $580 \cdot 579 = 335\,820$ combinations of time series obtained from not shaking devices conjointly as our negative class N. Note that we exclude pairs of same type of devices (two mobile phones as well as two smart watches) as these scenarios are not realistic in real life. The resulting data sets for the P and N class are used to evaluate the performance of the subsequently described constituent parts of ShakeUnlock.

6.5.2 *Performance Measures*

As the sizes of our P and N class differ notably, some performance measures like accuracy are not significant [200]. We therefore rely on a number of well known and more significant metrics in our evaluation. The true positive rate (TPR) represents the ratio of correctly identified cases of users trying to trigger an authentication state transfer with devices being shaken conjointly (P class samples). Likewise, the true negative rate (TNR) represents the ratio of correctly identified cases of minimal effort attacks, with devices not being shaken conjointly (N class samples). We obtain the TPR and TNR for all possible metric thresholds, from which we construct the receiver operating characteristics (ROC) and the area under the ROC curve (AUC). Both ROC and AUC capture the overall performance instead of stating the performance at a specific metric threshold. The equal error rate

(EER) states the error for TPR = TNR, representing the intersection between the ROC curve and the diagonal from TPR = TNR = 1 to TPR = TNR = 0.

6.5.3 *Magnitude Squared Coherence with Acceleration Time Series Magnitudes*

With magnitude squared coherence [363] the time series x and y are divided into n overlapping slices (Fig. 23). Each slice is multiplied with a weighting window (such as a Hann or Hamming window). We use slices of $\frac{7}{8}$ overlap and 1 s duration (with 100 Hz sampling rate this corresponds to slice and window lengths of 100 samples), and a Hann weighting window as proposed in [224]. Next, all slices are transformed into the frequency domain by applying a standard fast Fourier transformation (FFT) with 1 s window size. For each pair of corresponding slices from x and y, the coherence vector $C_{xy,n}(f)$ is calculated from the power spectral densities $S_{xx,n}$ and $S_{yy,n}$ and the cross spectral density $S_{xy,n}$ (Eq. 12). Then, all n coherence vectors $C_{xy,n}(f)$ are averaged to the single coherence vector $C_{xy}(f)$ (Eq. 13).

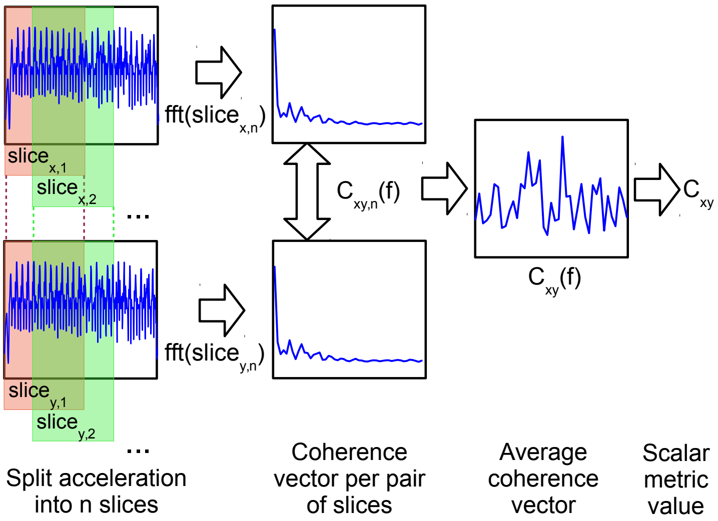


Figure 23: Active segment similarity analysis in ShakeUnlock.

$$C_{xy,n}(f) = \frac{|S_{xy,n}|^2}{S_{xx,n} \cdot S_{yy,n}} \tag{12}$$

$$C_{xy}(f) = \frac{1}{n} \cdot \sum_{n} C_{xy,n}(f) \tag{13}$$

Finally, a scalar metric value C_{xy} is obtained from $C_{xy}(f)$ using a collapsing function (Eq. 14).

$$C_{xy} = \mathrm{Col}(C_{xy}(f)) \tag{14}$$

This metric value C_{xy} is interpreted as confidence that devices have actually been shaken conjointly while recording x and y. Hence, if

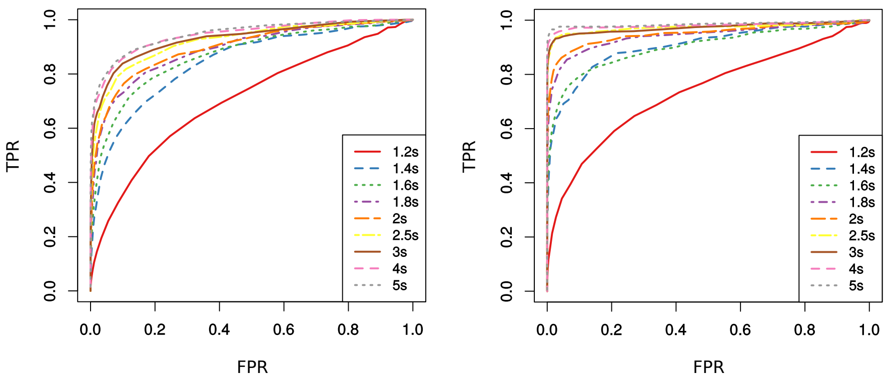
$C_{xy} \geqslant T$, with T being a predefined metric threshold, we transfer the authentication state and unlock the device. If $C_{xy} < T$ we refuse to transfer the authentication state, leaving the device locked. We apply the method as summarized above on the time series magnitudes of the two active segments x and y. Using the magnitude acceleration time series is done frequently to compensate for unknown spatial alignment of accelerometers. Thereby, time series magnitudes are calculated from the L^2-norm of the active segment 3D acceleration time series. As collapsing function Col we average the coherence vector $C_{xy}(f)$ up to a cutoff frequency of 40 Hz (Eq. 15).

$$C_{xy} = \frac{1}{41} \cdot \sum_{f=0\,\text{Hz}}^{40\,\text{Hz}} C_{xy}(f) \tag{15}$$

Using only magnitude squared coherence with acceleration time series magnitudes, we obtain an AUC of 0.8990 and an EER of 0.1777.

6.5.3.1 *Impact of Shaking Duration and Devices Being Apart From Each Other*

Results show that increasing shaking durations decreases overall error rates – for devices being held in the hand and strapped to the wrist, as well as devices being pressed against each other (Fig. 24).



(a) Devices at hand and wrist (b) Devices pressed against each other

Figure 24: ROC curves for different durations of users shaking their device, with (a) the devices being strapped to the wrist and held in the hand using ShakeUnlock data and (b) being pressed against each other in one hand using the database of [224].

Using a shaking duration of 2 s – which we assume is just short enough for users to consider shaking as an unlocking approach – we obtained an EER of 0.176 and a TPR/TNR of 0.795 and 0.867, respectively. These rates assume that both devices are shaken concurrently. Consequently, attackers trying to unlock the mobile phone which they previously got under their control have to perform this attack in parallel to users shaking their wrist watch accordingly. Further, the unlocking security level can easily be raised for users willing to shake their device longer (which could be chosen per user and application individually).

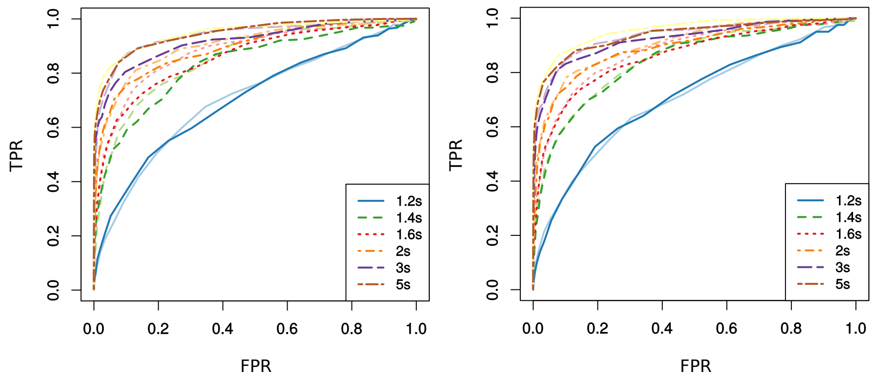
Using data of devices being pressed against each other for 2 s of shaking, we obtain an EER of 0.100 and a TPR/TNR of 0.885 and 0.925, respectively (Fig. 24b) – which is observably better over devices being apart form each other. These results support the intuition that the closer devices are together, the harder it is for an attacker to trick the approach into unlocking the mobile phone using non-correlated shaking. Furthermore, this suggests that an attacker being able to attach an acceleration sensor at the user (e.g. in clothing) will not be able to make immediate use of recorded acceleration data, except for when the acceleration sensor is very close to the wrist or hand, as the recordings will differ too much from the actual device acceleration.

Mayrhofer and Gellersen [224] report a TPR and TNR of 0.99 and 1, respectively, when employing the shake well before use database. These differences to our current findings are caused by utilizing a different threat model and differently sized negative classes C_N (time series used to compute the TNR). To obtain the TNR, the earlier evaluation uses a small dataset of 177×2 time series recorded by shaking devices simultaneously, but not with the same hand. Based on the resulting 177 time series comparisons, the TNR is computed. In contrast, in our evaluation we utilize the same dataset to obtain the TNR as well as the TPR: we compare all time series not recorded by shaking devices simultaneously with the same hand to compute the TNR.

6.5.3.2 *Impact of Sitting/Standing and Shaking with the Dominant/Non-Dominant Hand*

Figure 25 shows the impact of shaking devices with the dominant and the non-dominant hand as well as sitting or standing while shaking the devices based on our database.

It is clearly visible that shaking devices with the dominant hand (represented by the brighter lines in the left graph) with an EER of 0.168 and a TPR/TNR of 0.811/0.870 for 2 s of shaking consistently causes lower error rates compared to shaking devices with the non-dominant hand (represented by the darker lines) with an EER of 0.184 and a TPR/TNR of 0.779/0.863. We assume this to be the result of users shaking the devices slightly harder and/or faster as well as keeping the wrist more stiff – therefore causing more similar acceleration time series on both devices. Similar to using the dominant or non-dominant hand, sitting while shaking devices seems to cause slightly lower error rates compared to standing – with sitting (represented by the brighter lines in the right graph) causing an EER of 0.176 and a TPR/TNR of 0.806/0.866 for 2 s of shaking compared to



(a) Dominant and non-dominant hand

(b) Standing and sitting

Figure 25: Devices being shaken for different durations with (a) the non-dominant hand (dark) and with the dominant hand (bright) as well as (b) when standing (bright) and sitting (dark).

standing (represented by the darker lines) causing an EER of 0.177 and a TPR/TNR of 0.818/0.828.

To summarize these first findings: we applied magnitude squared coherence as demonstrated in [224] to data of devices somewhat apart and with non-static joint in between during shaking. Our findings support the intuition that increasing the shaking duration improves accuracy when assessing whether devices have been shaking conjointly, but obviously impair usability as the effort increases. Our findings further show that shaking devices with the dominant instead of the non-dominant hand or while standing instead of sitting results in slightly better accuracies, but overall has little impact. From those results we derive a shaking duration of 2 s to be a reasonable trade-off between usability and security. Consequently, for the subsequent evaluation of the constituent parts of ShakeUnlock we restrict ourselves to a shaking duration of 2 s. We therefore use active segments of 2 s duration per time series recording. Active segments shorter than 2 s are excluded from further analysis, as this simulates users not shaking their devices long enough.

6.5.4 *Optimal Time Series Derotation*

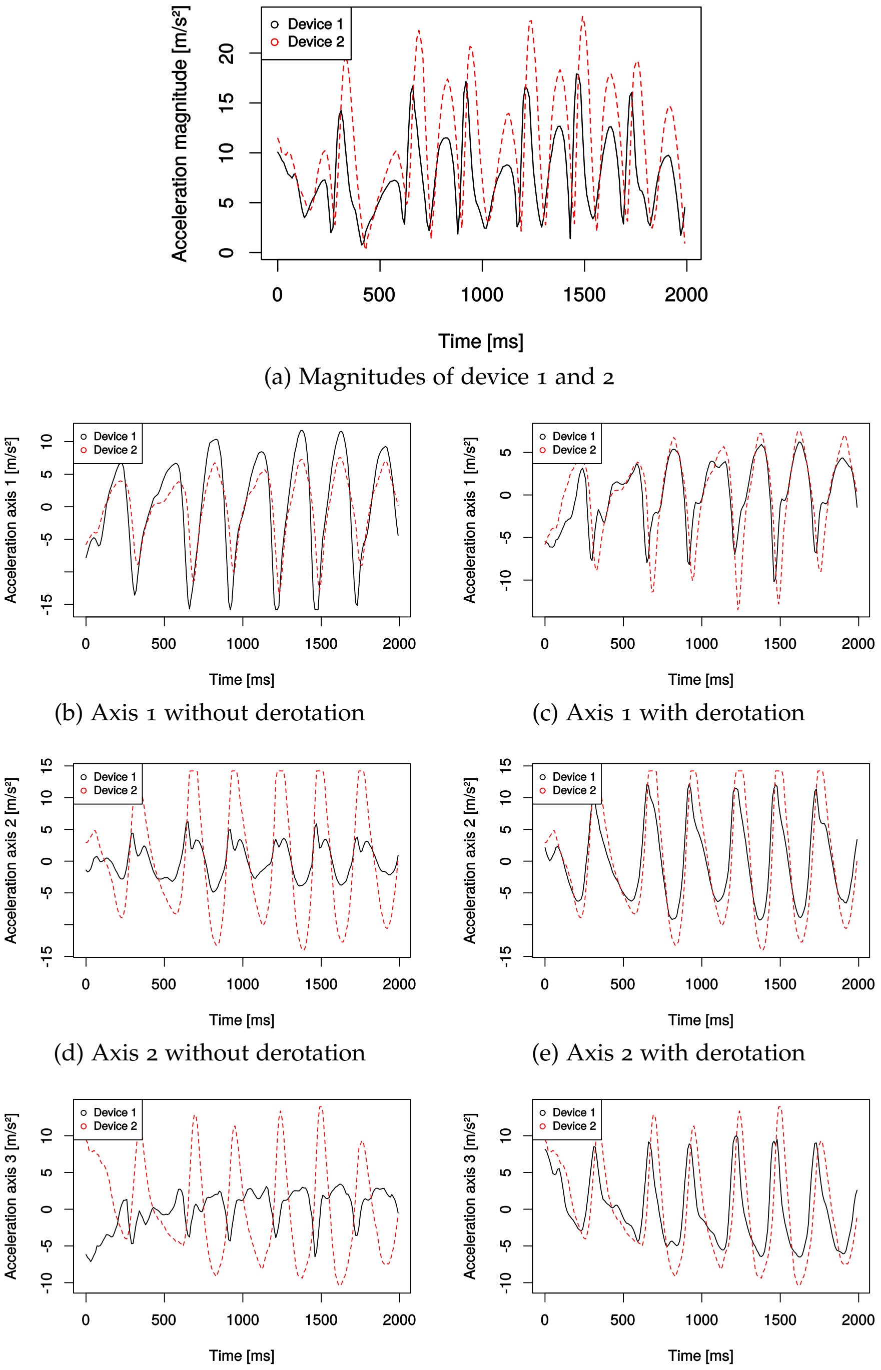
Most research on shared movement using records of 3D acceleration time series from mobile devices has focused on comparing acceleration magnitudes. This is because in general both the orientation of devices potentially moved together as well as the orientation of accelerometers within those devices is unknown. Unknown sensor orientation leads to axes of recorded time series not being spatially aligned, meaning they cannot directly be compared to each other. By calculating the magnitude this problem is circumvented, as magnitudes do not capture orientation information – thereby can directly be compared, even with unknown orientation of devices and sensors. This also causes the downside that not using orientation for comparing acceleration time series between different devices implies losing some potentially important information in the form of rotational components during the movement.

To allow for a meaningful comparison of two 3D time series in all their dimensions, the coordinate system of one 3D time series has to be derotated to suit the coordinate system of the other. This requires that both coordinate systems have retained their relative orientation throughout the shared movement of the devices (i.e. that rotations have been applied to both devices alike).

We have shown previously that a quaternion based approach can be used to analytically find the optimal derotation of two 3D time series, and that this improves the subsequent EER with various distance metrics [229, 230].

We now apply this quaternion based derotation as one constituent part of ShakeUnlock with the goal of improving overall authentication accuracy. Optimal derotation of two acceleration time series of devices being shaken can even be observed visually (Fig. 26, samples are taken from the ShakeUnlock database). As both samples are originated from devices actually being shaken together, the similarity between their time series is intended. This is well visible when comparing the time series magnitudes of both samples (Fig. 26a). Still, there are well observable differences between both samples when comparing individual axes – which originate from different orientation of devices and built in sensors (Fig. 26, left column). While the phase and periodicity of both samples seems to be comparable, the actual acceleration readings differ noticeably throughout the sample duration. Optimal derotation of one of the two samples results in better alignment of data of individual axes – which is visible in actual acceleration readings differing noticeably less than before derotation (Fig. 26, right column, with the sample of device 1 being derotated to match the sample of device 2).

When applying optimal derotation to our evaluation, Fig. 27 states the coherence density over frequency for the P and N class for applying coherence on magnitudes as well as on all axes of previously derotated time series. Brighter areas represent lower coherence, darker areas represent higher coherence. Coherence is observably more dense for the P class when derotating time series before computing coherence instead of computing the series magnitudes (Fig. 27a and 27b). In contrast, the density for the N class is only marginally influenced by derotating time series before computing coherence by being slightly higher on average (Fig. 27c and 27d). This is to be expected, as correlated time series initially are rotated arbitrarily but intentionally contain similarity – which causes derotated time series to show notice-



(a) Magnitudes of device 1 and 2

(b) Axis 1 without derotation

(c) Axis 1 with derotation

(d) Axis 2 without derotation

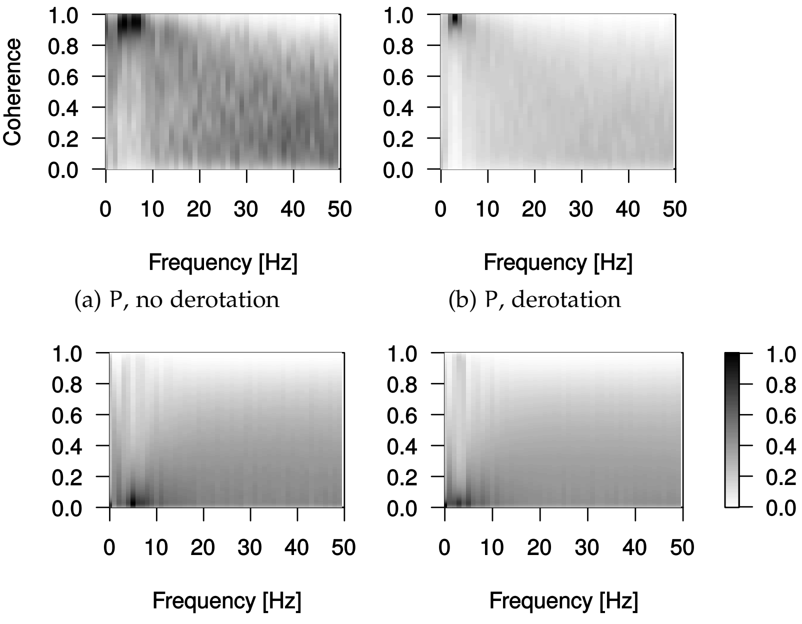
(e) Axis 2 with derotation

(f) Axis 3 without derotation

(g) Axis 3 with derotation

Figure 26: Sample 3D acceleration time series for two mobile devices being shaken together, depicted by their magnitudes (a) and by their individual axes without derotation (b, d, f) and with derotation (c, e, g), as demonstrated in [230].

ably higher similarity. In contrast, initially not correlated time series only have little coincidental similarity. Optimally rotating them therefore only causes an insignificant raise in similarity. This data suggests that for frequencies showing condensed coherence values, derotation of time series will improve class separation performance – which is supported by evaluation results stated below as well.



(a) P, no derotation

(b) P, derotation

(c) N, no derotation

(d) N, derotation

Figure 27: Coherence densities per frequnecy of P and N class without and with time series derotation.

In contrast to comparing time series magnitudes we instead compute coherence for each pair of axes (which have been aligned through derotation). Therefore, coherence computation yields three separate coherence vectors, one per (aligned) pair of axes. Each coherence vector represents the frequency range 0-50 Hz for 100 Hz sampling in data recording. Hence, all successive operations (e.g. filtering frequencies by applying a 0-20Hz bandpass) have to be applied to these three coherence vectors individually. We apply the previously used 40 Hz cutoff to the coherence vectors, then average them to obtain a final, scalar coherence value. By adding initial time series derotation to our evaluation setup, we obtain an AUC of 0.9214 and an EER of 0.1562.

6.5.5 *Coherence Frequency Bandpass*

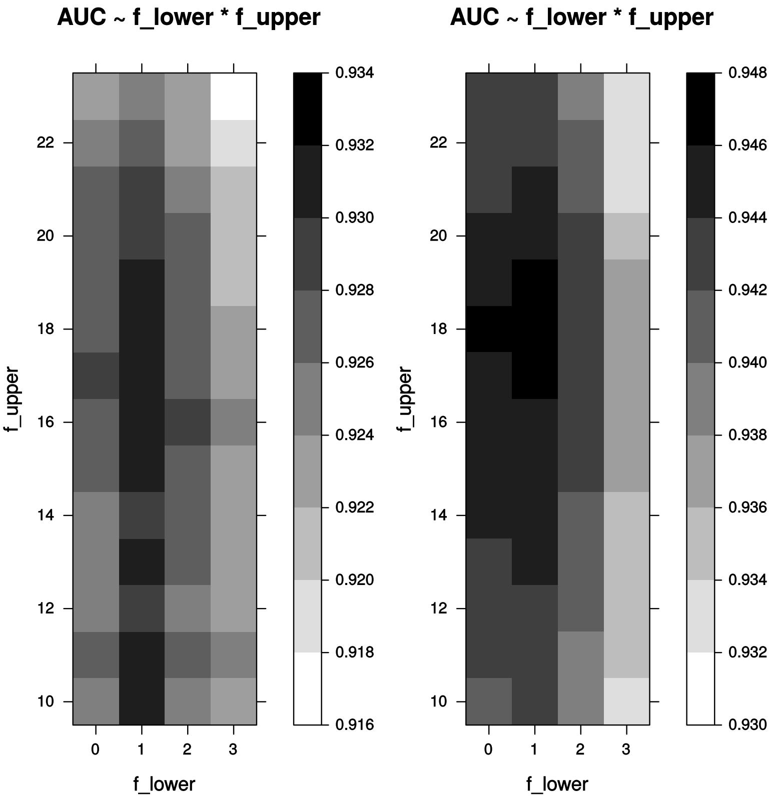
Overall, research on human body motion states quite different motion frequencies to usefully represent motion information. For example, in Biomechanics and Motor Control of Human Movement, Winter [368] states human body motion is in general represented by a frequency range of about 0-10 Hz. In contrast, e.g. Bouten et al. [40] find frequencies up to 20 Hz being useful to represent human movement during everyday activities. They further state that body movement of e.g. limbs is usually faster, compared to movement of torso and hip, whereas shaking mobile devices with the hand corresponds to the mentioned faster movements.

In their research on shaking devices conjointly, Lester et al. [198] pick up the frequency range of 0-10 Hz stated by Winter [368]. They average coherence in the range of 0-10 Hz to come up with a scalar similarity value. In contrast, Mayrhofer and Gellersen [224] average coherence in the range of 0-40 Hz to determine if devices were shaken conjointly without stating details on how this cutoff frequency was determined. It can be assumed that results from using a coherence range of 0-40 Hz were superior to results from using a range of only 0-10 Hz for their approach, for which the wider frequency range was used. To determine the optimal coherence frequency range we explicitly study the influence of different bandpass filters to classification performance.

As shown in the coherence distribution over frequency (Fig. 27), coherence is unequally distributed over frequency in the ShakeUnlock database. Overall, coherence is less dense as well as less diverse across P and N class for higher frequencies, compared to lower frequencies, although the lowest frequencies in the range of 0-2 Hz are less dense and less diverse across classes as well.

In order to utilize the best performing coherence frequency range in ShakeUnlock, we apply a bandpass to coherence frequencies before successively computing a scalar similarity value from the coherence vector. For real world applications and from an implementation point of view, using a bandpass has several advantages over more complex approaches of restricting the frequency range. Using a bandpass is intuitive and easy to understand. Further, it is fast and easy to implement and of small complexity. In our bandpass evaluation, f_L represents the lower frequency threshold, hence the lowest coherence frequency included during successive processing. Likewise, f_H represents the upper frequency threshold. The frequency bandpass performance (Fig. 28) states AUC over pairs of f_L and f_H, with darker areas representing higher AUC values, therefore better performance.

Note that with our setup, performance decreases notably when increasing f_L, while changes of f_H seem to have significantly less influence on performance. On the one hand, this indicates that the most important portion of information is contained in lower frequencies, and that higher frequency information is less reliable – which is in support of findings from previous research. If these lower frequencies are excluded, performance decreases significantly. On the other hand, including frequencies up to about 20 Hz can improve performance, which is different to what previous research would suggest [368].



(a) On magnitude data

(b) On derotated data

Figure 28: Coherence bandpass performance (AUC per bandpass filter setting) when applied without (a) and with derotation of time series (b). Note that left and right brightness scaling is differently to increase distinguishability.

With applying a bandpass to coherence frequencies from magnitudes of acceleration time series, performance peaks at $f_L = 1\,\mathrm{Hz}$ (skipping the 0 Hz constant component) and $f_H = 16\,\mathrm{Hz}$, with an AUC of 0.9315 and an EER of 0.1418. When combining the bandpass with initially derotating time series, peak performance is reached with consistent $f_L = 1\,\mathrm{Hz}$ and a slightly higher $f_H = 18\,\mathrm{Hz}$, with an AUC of 0.9469 and an EER of 0.1293. These results point out that coherence frequency range noticeably influences overall performance – and therefore should be selected carefully. In comparison to other constituent parts of ShakeUnlock using a coherence frequency bandpass turns out to hold the highest performance gain – while being amongst those easiest to implement.

6.5.6 *Coherence Frequency Collapsing Function*

In previous research on shaking devices conjointly, collapsing a coherence vector to a scalar coherence value has only been done by averaging coherence. To collapse a coherence vector, other functions are possible as well, with some of them being frequently used in other disciplines. We evaluate the following collapsing functions for obtaining a scalar similarity value from coherence vectors: sum (average), median, max, euclidean distance d_e, and square root distance d_s. Square root distance (Eq. 16) is the counterpart to euclidean distance (Eq. 17), by inversing the order of squaring and taking the square root. Additional functions such as min turned out to cause significantly worse performance in preliminary tests and therefore were disregarded in this evaluation.

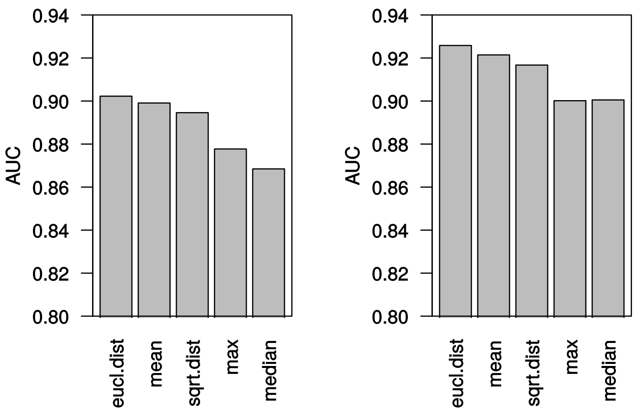
$$d_s(v) = \left(\sum_i \sqrt{v_i}\right)^2 \tag{16}$$

$$d_e(v) = \|v\| = \sqrt{\sum_i {v_i}^2} \tag{17}$$

Performance comparisons (Fig. 29) show euclidean distance slightly outperforms averaging as well as all other tested functions when used to collapse coherence vectors to a scalar similarity value for both time series magnitudes as well as initially derotated time series.

When applying euclidean distance as the best performing collapsing function to coherence obtained from time series magnitudes, we obtain an AUC of 0.9023 and an EER of 0.1670. In contrast, when applying euclidean distance as collapsing function conjointly with initially derotating time series and using a coherence frequency bandpass filter we obtain slightly reduced performance, with an AUC of 0.9464 and an EER of 0.1293.

On the one hand, these findings indicate that obtaining a scalar coherence value from a coherence vector might be improved by con-



(a) Time series magnitudes

(b) Derotated time series

Figure 29: Influence of coherence vector collapsing functions on overall performance using (a) time series magnitudes and (b) initially derotated time series.

sidering not only the mean, but alternative collapsing functions such as euclidean distance. On the other hand, when used with other constituent parts of ShakeUnlock the performance gain is minor (or as in our case, performance even decreased slightly).

6.5.7 *Optimal Coherence Threshold per Frequency*

6.5.7.1 *Determining Optimal Coherence Thresholds*

After deriving a scalar similarity value from a coherence vector (obtained from two acceleration time series of devices shaken conjointly) usually one fixed threshold is used to separate the P and N class, as reported by Lester et al. [198] and Mayrhofer and Gellersen [224]. Using a single coherence threshold has a significant drawback: all frequencies are combined within one scalar value, therefore the threshold can only address all frequencies at once. Another approach is to use an individual and independent threshold for each coherence frequency. Each such threshold represents the optimal separation between P and N class for that coherence frequency – hence provides better class separation on individual frequency level. Optimal thresholds differ when derived from either time series magnitudes or from initially derotated time series as derotation changes coherence values (see example in Fig. 30). Fig. 31 states the optimal coherence threshold per frequency for using time series magnitudes as well as for incorporating initial time series derotation.

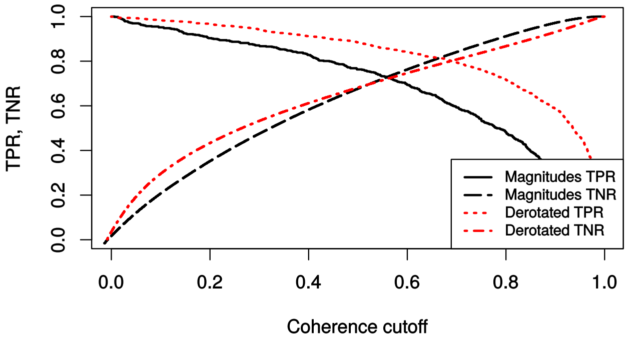


Figure 30: True positive and true negative rate over coherence threshold for 3 Hz. Match rates as well as coherence values themselves for 3 Hz are higher with derotation than with time series magnitudes.

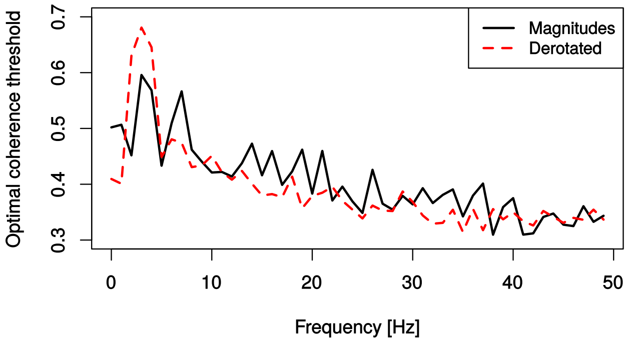


Figure 31: Optimal coherence thresholds per frequency.

6.5.7.2 *Using Optimal Coherence Thresholds*

Next, we determine if a coherence vector $C_{xy}(f)$ obtained by shaking device x and y corresponds to the P or N class using the optimal coherence thresholds $C_o(f)$. We have explored two ways of doing so: using a) a majority vote and b) the distances from the optimal thresholds. With the majority vote, we utilize the amount of frequencies being above their corresponding optimal threshold. If that amount is above another predefined threshold, the sample is classified as positive (shaken conjointly). If it is below the threshold, it is classified as negative (not shaken conjointly). In preliminary tests, the majority vote turned out to perform slightly worse than averaging the coherence vector.

We therefore incorporate the distance $d_{xy}(f)$ from optimal coherence thresholds $C_o(f)$ to coherence vector $C_{xy}(f)$ as well (Eg. 18). Its fundamental idea is that certainty rises with the distance to the corresponding optimal threshold. The larger the distance of a coherence value to its corresponding threshold, the higher the certainty that it belongs to the P respectively N class. To obtain a scalar similarity value from $d_{xy}(f)$, a collapsing function is required again. As with the previous collapsing functions evaluation (Sec. 6.5.6), once more euclidean distance slightly outperformed averaging the vector as well as all other collapsing functions (Eq. 19). Note that standard euclidean distance is not applicable anymore as it eliminates the sign for individual distances. We therefore use a signed euclidean distance $d_{es}(v)$ which preserves the sign of its components (Eq. 20 and 21).

$$d_{xy}(f) = C_{xy}(f) - C_o(f) \tag{18}$$

$$d_{xy} = d_{es}(d_{xy}(f)) \tag{19}$$

$$d_{es}(v) = a(v)^0 \cdot \sqrt{\mathrm{abs}(a(v))} \tag{20}$$

$$a(v) = \sum_i v_i \cdot \mathrm{abs}(v_i) \tag{21}$$

When incorporating the distance to the optimal coherence thresholds and signed euclidean distance collapsing with coherence obtained from time series magnitudes, we obtain an AUC of 0.9056 and an EER of 0.1724. When instead using it with initially derotated time series and using a coherence frequency bandpass filter, we obtain an AUC of 0.9495 and an EER of 0.1257.

6.5.8 *Coherence Frequency Weighting*

6.5.8.1 *Weighting Frequencies Individually*

The coherence density over frequency (Fig. 27) shows that coherence is denser for lower frequencies, with P and N class being visually more separated than with higher frequencies. Consequently, lower frequencies will yield better class separation performance than higher frequencies. Performances measures from classifiers using only a single coherence frequency to separate P and N class support this intuition with lower frequencies in general yielding better results than higher frequencies (Fig. 32).

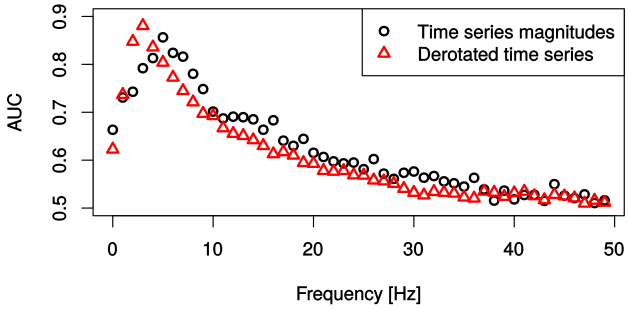


Figure 32: AUC of classifiers using a single frequency with and without derotation.

Note that without derotation (using time series magnitudes), the best performing frequency is 5 Hz. With derotating time series, the best performing frequency is shifted to 3 Hz. This is a side effect of derotation, which uses the largest eigenvector of the quaternion rotation matrix (obtained from the time series correlation matrix). Obviously, derotation favors 3 Hz alignment which indicates that optimal derotation can be achieved when aligning time series around that frequency. The dominant frequency seems to be 3 Hz when derotation shaking acceleration time series. Although the majority of AUC values is lower with using time series derotation, overall performance is better with using derotation (Sec. 6.5.4). This indicates that the performance gain through best aligning lower frequencies (increasing their corresponding performance) is higher than the performance loss through concurrently decreasing higher frequency performance. This underlines the importance of lower frequencies for separating P and N class (note the strong performance gain for 2 and 3 Hz). Moreover, this is in line with our previous finding of the best performing bandpass covering a narrower range of 1-18 Hz respectively 1-16 Hz, discarding higher frequencies.

From these insights it can be concluded that individually weighting coherence frequencies (e.g. based on their class separation power) when obtaining a scalar similarity value should improve results. The coherence frequency bandpass – as a less powerful, special case of such weighting – already showed to improve performance. With the bandpass, blocked frequencies are assigned a weight of 0, whereas passing frequencies are assigned a weight of 1.

6.5.8.2 *Obtaining Coherence Frequency Weights*

With our setup we weight 51 coherence frequencies in the range $[0, 1]$. Assuming a coarse granularity of 0.1 (11 steps of size 0.1 in the range $[0, 1]$) results in a grid search space size of 11^{51} – which is too large for a simple parameter grid search. We instead utilize an evolution strategy (ES) [29] to find a heuristic estimate of the optimal coherence frequency weights. We use a $(1 + \lambda)$-ES with $\lambda = 10$ mutants, randomly initialized starting weights, an initial maximum mutation rate of 1 per generation and a maximum mutation rate reduction of 0.005 per generation. With each generation, all parameters are mutated, and we run 919 generations in total (corresponds to a final maximum-mutation of 0.01). To obtain reliable results we repeat the ES 100 times (for both using time series magnitudes as well as initially derotating time series) and use the best obtained weights. The heuristic estimate of optimal coherence frequency weights shows that there is a decline of weights with increasing frequency (Fig. 33) – however, the decline is throughout unsteady.

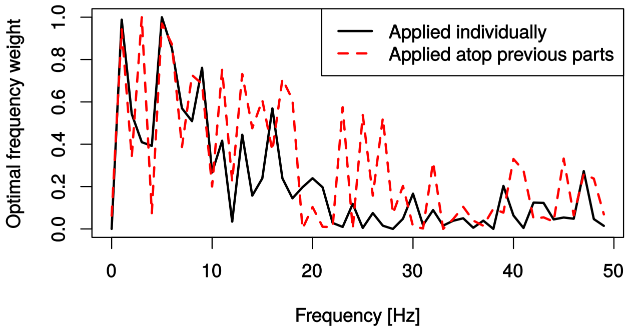


Figure 33: Heuristic estimation of problem specific, optimal coherence frequency weights.

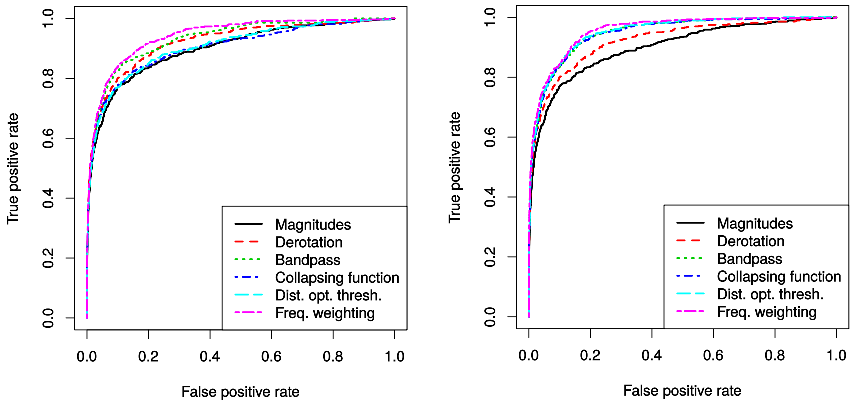
It is important to understand that these estimated weights represent a highly problem-adapted optimum of weights (overfitted to our problem) and therefore cannot be derived from discrimination power metrics like AUC or directly reused for problems without re-estimating the weights. Consequently, these weights just serve as a prospect of possible performance gain using frequency weighting and would have to be re-estimated if applied to other problems. Using the heuristic estimate of optimal coherence frequency weights on top of using time series magnitudes we are able to increase AUC to 0.9420 and decrease the EER to 0.1329. When instead applying it with initially derotating time series, using the distance to optimal coherence thresholds and euclidean distance as coherence collapsing function while replacing the coherence frequency bandpass filter, we are able to increase the AUC to 0.9551 and decrease the EER to 0.1258. These gains do not seem to outweigh the added complexity and risk of overfitting.

6.5.9 *Discussion of Performance Gain*

Note that the order of combining constituent parts influences the associated difficulty of achieving a performance gain (Fig. 34, Tab. 8). For constituent parts applied earlier more room remains to increase performance.

Constituent part	Implement. complexity	Individual		Atop prev. parts	
		AUC	EER	AUC	EER
Time series magnitudes (baseline)	low	0.8990	0.1777	—	—
Derotated time series	medium	0.9214	0.1562	—	—
Coherence frequency bandpass	low	0.9315	0.1418	0.9469	0.1293
Coherence vector collapsing fun.	low	0.9023	0.1670	0.9464	0.1293
Dist. to opt. coherence thresh.	medium	0.9056	0.1724	0.9495	**0.1257**
Coherence frequency weighting	high	0.9420	0.1329	0.9551	0.1258

Table 8: Contribution of constituent parts of ShakeUnlock to overall performance, applied individually and atop previous parts.



(a) Individual contribution

(b) Combined contribution

Figure 34: ROC curves stating (a) the individual contribution of constituent parts and (b) the combined contribution of constituent parts of ShakeUnlock to overall performance.

The highest performance gain is achieved by including coherence frequency weighting or its special case, the coherence frequency bandpass. This emphasizes the importance of carefully selecting coherence frequencies for human body motion analysis tasks. With frequency weighting, implementation complexity is worth mentioning: we use heuristically obtained estimates of optimal weights and these weights have to be re-estimated when applied to different problems. In contrast, the coherence frequency bandpass provides an easier to implement alternative to frequency weighting. It achieves optimal performance by including acceleration frequencies of up to about 20 Hz. This supports findings from previous research which suggest –

against common assumptions – that human body movement includes useful information up to or even beyond a frequency of 20 Hz.

The second highest performance gain is achieved using optimally derotated 3D acceleration time series in consecutive analysis instead of using acceleration time series magnitudes. Computing time series magnitudes strips out rotation information contained in original 3D time series. In contrast, with optimally derotated time series, parts of rotation information remain (namely changes in rotation over time), which is supported by improved performances. Consequently, derotation of 3D acceleration time series should be considered before doing consecutive analysis.

Including distance to optimal threshold and modified coherence vector collapsing functions achieve minor performance gains. With the first, the coherence threshold for separating classes is chosen optimally for each frequency. With the latter, euclidean distance turned out to slightly outperform the frequently used averaging of coherence on overall performance. When applied individually, both achieve a small performance gain. When applied in combination with derotated time series and a coherence frequency bandpass, their performance gain is negligible, hence – depending on the problem – they can be excluded from implementation in favor of frequency bandpass and optimal derotation of time series.

6.6 implementation and user study

Based on findings from our evaluation we implemented ShakeUnlock on Android for mobile phones and wrist watches[5]. In the implementation the link is established as soon one devices starts recording an active segment and acceleration recordings are aggregated on the mobile phone afterwards. In case one device did not detect an active segment, unlocking is aborted and the user is notified. Further, the user is notified about all successful or failed ShakeUnlock attempts on both mobile phone and smart watch. This ensures the user is informed in case case of the mobile phone being under control of an attacker. Based on our finding, for active segment similarity analysis we chose to include optimal derotation of 3D acceleration time series, applying a coherence bandpass filter and collapsing the remaining coherence vector to a single scalar value using euclidean distance.

Using our implementation we conduct a user study to quantify the impact of attacks on ShakeUnlock as summarized in Sec. 6.2, and to measure upper boundaries (which are expected to break unlock security). The study featured a total of 15 pairs of participants pairwise attacking each other 20 times per attack scenario (which results in a total of 600 attacks per scenario). For cooperative attacks, participants

[5] The ShakeUnlock implementation source is available via git clone from `git@bitbucket.org:usmile/shakedemo.git`.

were told to utilize any cooperative strategy or tool at hand except for touching the other device or participant. This lead to participants using verbal communication, music, or even a metronome as help for synchronization.

From study results, we found the FPR to be 0.20 for observatory attacks, 0.35 for cooperative attacks, and 0.90 for handshaking attacks (all with a threshold of 0.522, which corresponds to a TPR of 0.82 computed from ShakeUnlock database data only[6]). On the one hand – in contrast to [224] – in our setup, forging the second shaking pattern seems feasible with a rate of about 0.2. We infer that this is caused by the wrist as joint in between devices (instead of devices being pressed against each other) – which causes sensed acceleration to be different on devices when shaking them, consequently lowering the required similarity of acceleration records for unlocks as well as attackers. On the other hand, although this is a realistic attack, it is connected to a certain effort, as attackers are required to a) acquire an identical looking device and b) replace the user's phone with the proxy device. From study results, we further consider both cooperative and handshake attacks to break ShakeUnlock in terms of unlock security. We argue that this is acceptable, as we also consider them unrealistic/easily detectable in real life unlock situations.

6.7 SUMMARY

For token-based user-to-device authentication we propose ShakeUnlock to conjointly shake an unlocked, mobile token device and another mobile device still locked to transfer the authentication state from the token device to the other device and unlock it. A common use case features a wrist watch as token device strapped to the wrist and a mobile phone held in the same hand. Both are pre-paired and can communicate over a secure channel. While devices are shaken, we record 3D acceleration time series on both devices. These are analyzed for similarity to decide if both devices have actually been shaken conjointly. Therefore, shaking devices serves as secure trigger mechanism to transfer the authentication state. ShakeUnlock has the advantage of requiring only acceleration sensors, which are commonly integrated in mobile devices. Further, acceleration recording can be done power efficiently using dedicated hardware – similar to background step counting, which is already available in several off-the-shelf mobile devices from various OEMs. The evaluation of ShakeUnlock includes the influence of users using their dominant or non-dominant hands and sitting or standing, as well as the contri-

[6] The EER composed from one-vs-all comparisons using positive samples of the ShakeUnlock database and negative samples only from the observably attack study is slightly lower with 0.19; using cooperative attack data instead it is 0.23 and with handshake attack data it is 0.45.

bution of constituent parts to the system performance. We find that using the dominant hand or standing leads to slightly improved accuracies over using the non-dominant hand or sitting – but overall this seems to have little impact. In terms of contribution of constituent parts of ShakeUnlock we find coherence frequency filtering and optimal derotation of 3D acceleration time series to be most effective in improving the distinguishability of legitimate unlocks and potential attacks. We further implemented ShakeUnlock on off-the-shelf Android devices. Using live data from our implementation, 15 pairs of participants tried to attack each other and trigger unlocks in different attack scenarios. Results indicate that observational attacks have a success rate in the range of 0.2. This is higher than anticipated, but seems acceptable, as for this, attackers at first need to a) replace users' devices in secret with mock devices and b) need to shake the obtained device at the same time as users (with users being informed about unlock attempts), creating significant barriers for a successful attack.

We thereby conclude that ShakeUnlock is a mobile device unlock approach complementary to existing unlocking approaches (e.g. PIN, password, unlock pattern, or fingerprint). Similar to these it solves not all but parts of the problem of unlocking mobile devices during everyday usage. ShakeUnlock provides an additional option for performing unobtrusive mobile authentication in certain situations that users can choose to use. It thereby contributes to unobtrusive mobile authentication by addressing different situations in which authentication might be required compared to existing approaches (e.g. unlocking mobile devices one handedly without looking at the screen), not imposing cognitive load on users, and a duration in the range 2 s to perform the authentication state transfer. Future work on ShakeUnlock could investigate long term acceptance with an extensive usability study. Such a study needs to consider e.g. muscle memory effects, its learning rate, and effect on usability over time. A short study would likely only give limited insights and possibly be biased towards negative feedback as it might not be able to account for learning a muscle memory or related effects. Hence, this study should be performed longitudinally, spanning several weeks or months.

7 EMPLOYING VIBRATION FOR DEVICE-TO-USER AUTHENTICATION

In this chapter we highlight our vibration based device-to-user (D2U) authentication approach which communicates an authentication secret to users with a vibration code (Fig. 10, c). Parts of this chapter have previously been published in [105].

Attackers who obtain control over a mobile device cannot access data stored on it if the device is properly protected with a local physical access protection mechanism, e.g. that requires successful authentication before being unlocked. However, attackers can perform hardware phishing attacks to trick user into unwittingly revealing secret authentication information to an identically looking but malicious phishing device (Sec. 3.6.1). This information can be relayed to attackers who can possibly use it to perform authentication to the original device, thereby to access data processed and stored on it.

Employing mobile D2U authentication would be one way to impede hardware phishing attacks. Based on the little previous work in this field (Sec. 3.6) we at first discuss different possibilities to establish mobile D2U authentication suitable for our scenario. In order to provide a first countermeasure to hardware phishing attacks we then present a mobile D2U authentication approach using vibration patterns. Our approach communicates authentication information from mobile devices to their users with vibrations. The vibration pattern for a specific device is previously known to its users. When users hold the mobile device in their hands and D2U authentication is performed they should recognize the vibration pattern either as being genuine (indicating a higher probability that the device is in fact the genuine one) or as being different or missing (indicating that the device is probably a different one). Summarizing, the contributions of our mobile D2U authentication approach are:

- We provide an overview of possible D2U authentication approaches and compare their advantages and drawbacks for mobile devices, including estimated bandwidth and possible risks.
- We analyze vibration as one such D2U channel in detail, including the design of a vibration code consisting of different vibration patterns and its evaluation with a user study on how well those vibration patterns can be distinguished by mobile users.

Our approach to mobile D2U authentication thereby contributes to unobtrusive mobile authentication by providing a first step towards

	See	Hear	Feel
Visual	+	-	-
Sound	-	+	-
Vibration	-	o	+

Table 9: Possible D2U authentication approaches with strong (+), weak (o) and few/no correlation (-) with human sensing capabilities.

closing the currently unaddressed aspect of D2U authentication on modern mobile devices. In the future, D2U authentication employed on mobile devices can impede hardware phishing attacks, thereby provide a different aspect of protecting sensitive data on mobile devices from unauthorized physical access of third parties.

7.1 POSSIBLE WAYS OF DEVICE-TO-USER AUTHENTICATION

Combining capabilities of current mobile devices and human sensing, different D2U authentication approaches seem possible (Tab. 9). All of them could be employed standalone or merged into a single hybrid approach. Further, all of these could be used for the device revealing authentication information to the user before, during, or after the user authenticates to the device.

7.1.1 *Visual*

One obvious D2U authentication is to show authentication information visually, e.g. on the mobile device display. Notification elements could be used as well (e.g. the LED usually indicating the reception of messages or calls). While displays feature higher channel bandwidth, notification elements could show information even when the screen is off (which does not seem to be an advantage in terms of security). Similar to the concept of showing a secure authentication image to the user [288, 289], this approach is prone to shoulder surfing.

7.1.2 *Sound*

Analogous to using visual information, authentication information can be revealed via sound. For example, HAPADEP [319] uses a human recognizable MIDI codec transporting 240 bits of information in 3.4 s (~70 b/s), which seems sufficient for D2U authentication tasks. Similarly to visual approaches, sound is prone to attackers observing authentication information without physical access to the device.

7.1.3 *Vibration*

Information emitted by device vibrators can conceptually be observed by a) feeling the vibration and b) hearing noise caused by vibrators – given a quiet environment. In contrast to previous concepts, vibration cannot be visually observed by attackers, which is a valuable advantage in terms of security. It further is unobtrusive as users do not need to look at the screen or have to listen to sounds in a possibly noisy environment [4]. A drawback is attackers potentially being able to observe vibration pattern sounds in quiet environments. While this could be exploited to obtain secret information, it is likely still more complicated than e.g. overhearing authentication via dedicated sound or observing secret information displayed on mobile device screens via shoulder surfing. We are currently not aware of any research stating channel bandwidth of users distinguishing vibration patterns. This is, together with its favorable security properties, why we conduct a user study on evaluating how well preliminary vibration patterns can be recognized by users.

7.1.4 *Interlock Authentication*

For all mentioned possible D2U authentication channels, there exist multiple variants of how to integrate D2U authentication with U2D authentication. The first is to have the device authenticate to the user before the user authenticates to the device. On the one hand, this ensures users that it is the correct device they are revealing their authentication secret to. On the other hand, in case attackers get physical access to the device (without being aware of the user authentication secret, so they cannot unlock the device), they would be able to observe the D2U authentication secret – and could later mock it too, using a phishing hardware device. If instead the user authenticates to the device first, and afterwards the device to the user, hardware phishing attacks are possible, as the device only authenticates after the user authentication secret has been fully revealed.

A more promising variant would be using the interleaving "interlock" information exchange [178, 287] to integrate user-to-device and D2U authentication. Interleaving authentication information is well known and in active use in a variety of areas (e.g. to prevent different types of attacks on network communication and key exchange protocols [228]). Interleaving could start with the device revealing the first authentication part to the user, right before the user starts authentication to the device (e.g. when the screen is turned on). Successive parts would be revealed only if the user enters correct authentication information. Here, the difficulty could again lie with the human factor: users experience a potentially increased authentication effort and are required to stop entering further authentication information to the device, if the device does not reveal itself as their trusted device.

Summarizing, using vibrations seems better suited for D2U authentication than using visual information or sound. Though there exist several studies of M2M communication using mobile device vibration as communication channel, which state the channel bandwidth in the range of tens b/s [375] to hundreds b/s [4, 134, 294], we are not aware of any vibration channel bandwidth analysis that involves humans and devices (e.g. how much information a human can possibly extract from machine vibration patterns). Therefore, vibration could prove suitable for D2U communication – thereby also for D2U authentication, which we investigate in the next section.

7.2 THREAT MODEL

Without employing D2U authentication, attackers could use hardware phishing attacks to obtain authentication information from users who unwittingly authenticate to a phishing device (Sec. 3.6.1). Attackers can then use the obtained information to gain access to data stored on a user's device. Hardware phishing attacks are fostered by two factors: a) no countermeasures to hardware phishing attacks being employed with most modern mobile devices and b) hardware phishing attacks not requiring special knowledge, in contrast to the previously discussed threats of reconstructing biometrics from obtained biometric templates or forging shaking patterns from obtained shaking acceleration. For this reason hardware phishing attacks could be performed by a broader range of attackers. While hardware phishing attacks bring initial purchasing cost for obtaining an identically looking phishing device, attackers do not "spend" this cost but only exchange devices, which can be considered cost-neutral in the long view. Consequently, the effort of performing a hardware phishing attack is limited to observing which device and locking mechanism is used, obtaining and configuring a phishing device to look identically and to forward any entered information to attackers, and swapping the phishing device with real device while it is unattended.

When employing vibration based D2U authentication users observe an authentication secret their device communicates to them using vibrations. In case of attackers performing hardware phishing attacks without taking D2U authentication into account, users would notice that the device is not communicating any authentication secret. Users can therefore stop using the device (this includes aborting a potentially ongoing user-to-device authentication) and investigate the issue. In case attackers are aware that vibration based D2U authentication is employed but have no knowledge of the exact vibration pattern used they could choose to use a random vibration pattern with the phishing hardware. Users should thereby recognize that the vibration pattern they observe is different and abort device usage as in the previous case. The requirement that arises from this threat is that vibration patterns within a predefined vibration code for D2U authentication should be well distinguishable by users. In case attackers gain knowledge of the vibration pattern the device uses to authenticate to their users, they can employ the same pattern to perform hardware phishing attacks without alarming users during the attack. For this reason, like device-to-user authentication secrets, D2U authentication secrets should not become known to attackers.

Visual or sound based D2U authentication could easily be eavesdropped by attackers, e.g. by using shoulder surfing or being close to users and listening while they perform authentication. In contrast, obtaining vibration patterns is more complicated for attackers. If the device communicates the authentication secret before user-to-device authentication is performed, attackers could grab the device while its unattended to observe the vibration pattern. If the device performs D2U authentication after user-to-device authentication is performed attackers cannot observe the vibration pattern this way. However, obtaining it might not be necessary at all, as attackers could perform hardware phishing attacks without employing vibration based D2U authentication, because users do not expect any vibrations until user-to-device authentication has been performed. In contrast, if an interlock based authentication is utilized, attackers would need to obtain the D2U authentication secret and need legitimate users to perform user-to-device authentication for the D2U authentication secret to be revealed. As physically touching the device to observe its vibrations at the same time users are authenticating is unlikely to stay undetected, attackers are limited to eavesdropping the sound that mobile devices make while vibrating[1]. This improves the threat model in two ways. First, the time window for obtaining D2U authentication secrets is limited to when legitimate users authenticate. Second, while eavesdropping vibration patterns seems possible in quiet environments, it is presumably connected to an increased effort or impossible altogether in noisy environments. This might require attackers to use additional tools (e.g. microphones, amplifiers, and/or analysis of recorded audio signals). We argue that those additional steps increase the effort for attackers, thereby raise the bar for successfully performing hardware phishing attacks and improve the corresponding threat model.

[1] As with our previously discussed threat models we declare attackers using malware to eavesdrop and extract D2U authentication vibration patterns from mobile devices out of scope.

7.3 DEVICE-TO-USER AUTHENTICATION USING VIBRATION PATTERNS

In this section we design a vibration code consisting of different vibration patterns that mobile devices can use to communicate authentication secrets to their users. In order to estimate how well users would be able to correctly recognize such a code we further perform an according evaluation. In this evaluation, we measure how well users are able to correctly recognize a familiar vibration pattern and how well they are able to distinguish different vibration patterns – that possibly feel similar – from each other.

7.3.1 *Preliminary Vibration Code*

The main limitation of vibration for user friendly D2U authentication is duration: if authentication takes noticeably longer when incorporating device authentication, the vibration feedback will possibly not be employed by users. As mobile U2D authentication usually takes in the range of 1.5–3.5 s (depending on the employed unlocking approach) [150, 377], we restrict ourselves to a window of this size. For example, using a 4 digit PIN for user authentication with an estimated duration of 2 s would result in revealing the next digit to the device about every 0.5 s. This 0.5 s window could be used to reveal a part of the D2U authentication information via vibration. Based on these limitations and a preceding, preliminary study on which vibration types and timings are easy to be distinguished correctly, a prototypical vibration test code was derived. Consequently, with more in-depth insights to human vibration pattern recognition capabilities this code (and its bandwidth) could likely be improved.

Our preliminary vibration code consists of different vibration patterns. Each vibration pattern contains 1–2 groups of vibrations, with each group consisting of up to 3 single vibrations (Fig. 35). The second group is allowed to be empty (containing no vibrations), while the first group must contain at least one vibration. This results in our test code being able to transport one of a total of $3 \cdot 4 = 12$ different patterns per transmission. Vibration and pauses between vibrations of the same group are of 60 ms duration. Pauses between vibrations of different groups are of 200 ms duration. This setup results in an average pattern duration of 465 ms, which would be within the hypothetic 0.5 s time frame for feedback with a 4 digit PIN entered in 2 s – and which results in a bandwidth of ~7.7 b/s. Subsequently, we depict patterns as the amount of vibrations contained in each group, e.g. "3 2" for the first group containing 3, the second 2 vibrations, or "2" the first group containing two vibrations and the second being empty.

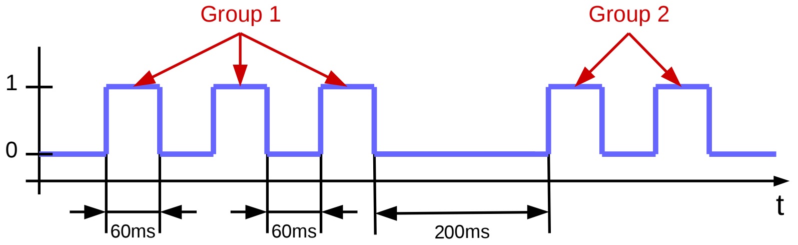


Figure 35: Example vibration pattern "3 2" from our preliminary vibration code, with 0 and 1 indication no vibration and vibration, respectively.

7.3.2 *Vibration Pattern Recognition Study Setup*

The preliminary vibration code has been implemented in an Android application[2] for the successive user study. The application features two modes: in trial mode, users can trigger all different vibrations as they wish and learn how they feel. In test mode, users are assigned a randomly chosen vibration pattern and have to decide for further, also randomly chosen vibration patterns, if this was their assigned pattern.

12 people participated in the study and were allowed to try out the application in trial mode as long as they wished. Each participant did at least 12 vibration pattern recognition sets in test mode, where for each test set they were assigned a random pattern and had to decide for 16 further random patterns (which they could trigger only once), if it was their assigned pattern. The probability of the test pattern being the assigned pattern was set to $\frac{5}{16}$. This setup resulted in 898 and 1614 recognitions of assigned and non-assigned patterns, respectively[3].

7.3.3 *Vibration Pattern Recognition Results*

Vibration pattern recognition rates over all users (Fig. 36a) indicate that our vibration patterns can successfully be distinguished. There seems to be no trend of shorter or longer patterns being recognized correctly with higher probability. Instead, recognition correctness involving vibration patterns "2", "1 1" and "2 2" seems to be lower, compared to recognition not involving these patterns.

The confusion of recognition correctness for all possible combinations of assigned and presented patterns (Fig. 36b) and the distribution of true positive and true negative recognition rates for all vibration patterns (Fig. 36c and 36d) indicate that if users are presented their assigned patterns they can likely recognized it correctly, with a

[2] The application code is open source and publicly available at https://github.com/mobilesec/device-to-user-authentication-vibration-bandwidth.

[3] Detailed study results are publicly available at https://www.usmile.at/downloads/.

median correctness of 97.5%. This further indicates that non-similar patterns are even likelier correctly recognized as being different, e.g. with patterns such as "1" and "3 3", which have been distinguished without a single error. But this also indicates that there is a tendency of users to incorrectly recognize non-assigned patterns as their assigned ones, if patterns are similar. For example, pattern "2" and "2 1" have frequently been mis-recognized as "1 1" (error rates of 27% and 9%), pattern "1 3" as "1 2" (15%), pattern "2 3" as "1 3" (14%), or pattern "3 2" as "3 3" (20%). The resulting median recognition rate over all assigned and non-assigned patterns is 97.5%.

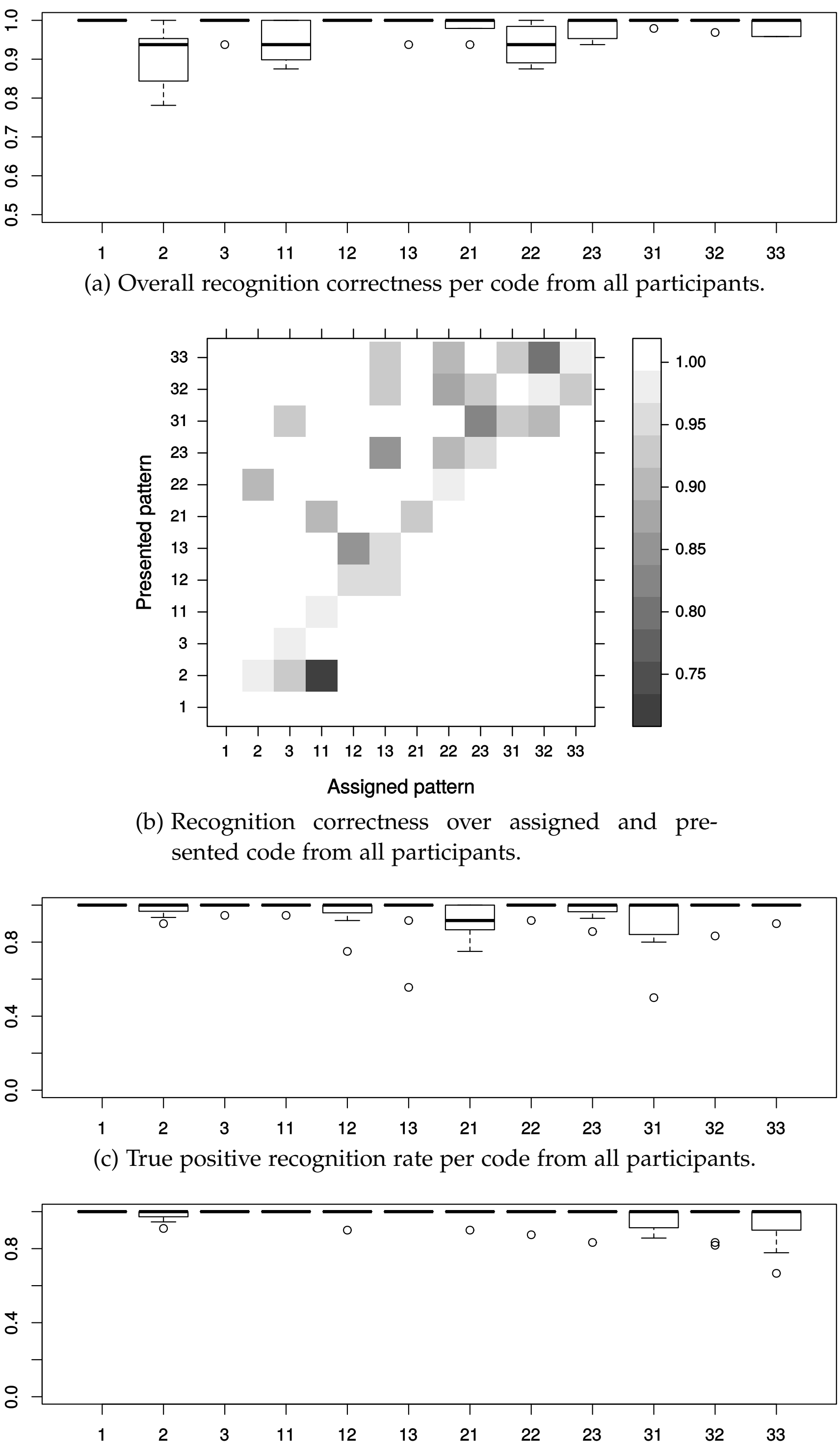
Despite these errors our preliminary vibration code has an average bandwidth of ~7.7 b/s and our results show a median successful vibration pattern distinguishing rate of 97.5%. From this we infer that vibration patterns could serve as valuable D2U authentication channel.

After finishing the study about 50% of participants stated that they believed they used hearing vibration patterns in combination with feeling them to decide if it was their assigned pattern. This indicates that hearing and feeling are used together for recognizing vibration patterns. Consequently, future research should investigate human vibration pattern recognition capabilities from only feeling patterns (e.g. with suppressing vibration sounds for participants or having them listening to music) as well as from only hearing patterns. Although the latter represents the scenario of attackers possibly being able to overhear secret vibration authentication information we argue that this is likely still more complicated than e.g. overhearing dedicated sound or observing secret information displayed on mobile devices via shoulder surfing.

7.4 SUMMARY

Our approach towards vibration based D2U authentication shows promising results, with vibration patterns – that act as authentication secrets in our scenario – being recognized correctly with a median accuracy of 97.5%. Further, results confirm intuition that patterns observed as being more similar to each other also seem harder to be distinguished correctly. Our findings thereby indicate that vibration in the future could be utilized as unobtrusive and potentially hard-to-eavesdrop D2U authentication feedback channel.

Within the context of our overall goal D2U authentication thereby addresses users being able to unobtrusively recognize their devices not only by their appearance but also by an authentication secret devices communicate back to them. This impedes attackers performing hardware phishing attacks, thereby also contributes to preventing unauthorized physical access of third parties to data processed and stored on modern mobile devices. D2U authentication could



(a) Overall recognition correctness per code from all participants.

(b) Recognition correctness over assigned and presented code from all participants.

(c) True positive recognition rate per code from all participants.

(d) True negative recognition rate per code from all participants.

Figure 36: Participants' recognition rates of vibration patterns as (a) overall recognition correctness per codes involved, (b) distribution of recognition correctness over assigned and presented codes, (c) distribution of true positive recognition rates per code, and (d) distribution of true negative recognition rates per code.

thereby be performed in parallel to users authenticating to devices themselves – which would be less obtrusive in terms of required time dedicated to D2U authentication. In combination with an anticipated, possibly arising muscle-memory-like effect this would lead to users intuitively recognizing their device by the familiar vibration pattern – without explicitly concentrating on it. Vibrations differing from the known pattern would lead to users becoming suspicious and either aborting their own ongoing authentication to the device or at least being able to initiate countermeasures if they already performed authentication to limit possible damage. The combination of unobtrusiveness and difficulty for attackers to eavesdrop on the secret – in comparison to visual or audio based secrets – thereby incorporates to our overall goal of providing additional unobtrusive security in form of authentication for everyday usage with mobile devices.

Future research on vibration based D2U authentication could investigate eavesdropping resistance. Within our evaluation participants stated they used hearing too to decide if a vibration patter was their assigned pattern. Consequently, future research should investigate, amongst others, human vibration pattern recognition capabilities by only hearing or feeling them – with the latter representing a possible attack scenario of attackers in quiet environments observing vibration authentication information by hearing it. This would aid the design of robust and distinguishable vibration patterns as well as secure exchange of information between users and devices based on vibration in general. While we evaluated how well vibration patterns can be distinguished, we leave the evaluation of muscle-memory like effects for future work. Future work investigating if and how such muscle-memory-like effects arise with intuitively distinguishing vibration patterns and correctly recognizing a specific pattern would need to be performed longitudinal, e.g. with daily usage over weeks or months. Further, future research could investigate other possibilities than using visual, audio, or vibration communication of information from devices to users. The required core aspects such communication for the purpose of authentication are twofold: a) being unobtrusive to users by e.g. being easy to remember, distinguish, and preferably possible to be done in parallel to users' activities on the mobile device, and b) being difficult for attackers to observe to protect the authentication secret.

8 RECAP OF OUR APPROACH FROM AN ATTACKER'S PERSPECTIVE

In this chapter we review our approach from an attackers perspective. We recap the situation without using our approach, including threats and possible attack scenarios and discuss the general potential for optimization in order to impede those attacks and to improve the threat model. We then discuss which threats are addressed and in which ways possible attacks are impeded. We also discuss which threats remain or have arisen anew with our approach – which consequently remain for further investigation in future work.

8.1 AN ATTACKER'S PERSPECTIVE ON MOBILE AUTHENTICATION WITHOUT OUR APPROACH

8.1.1 *Mobile Device Authentication Without our Approach*

When users authenticate and interact with their mobile devices without using our approach, from a top-level view of the attacker's perspective there exist several possibilities to e.g. obtain physical access to mobile devices, the authentication secret/token, or users' biometrics (Fig. 37).

Knowledge based authentication on mobile devices bears cognitive load on users that increases with the length and complexity of the authentication secrets as well as the amount of devices to protect. The input of secrets for authentication on mobile devices can be cumbersome and time consuming due to small user interfaces and little haptic feedback and further requires user attention (e.g. having to look at the screen). These drawbacks are known to cause users to choose weak secrets or even not use knowledge based authentication at all. From a threat model perspective this leads to certain mobile devices being unsecured or protected with weak secrets – which aids attackers in physically accessing unsecured mobile devices or guessing weak secrets using brute force approaches. Further, the input of knowledge based secrets can be observed by attackers using shoulder surfing or smudge attacks and used with replay attacks to access mobile devices. Biometrics based authentication does not bear cognitive load on users, therefore can be considered to be less obtrusive. However, biometrics cannot easily be exchanged in case of their disclosure to third parties. From a threat model perspective, when biometrics are used on mobile devices for unobtrusive authentication, they consequently might themselves become a high value target for attackers.

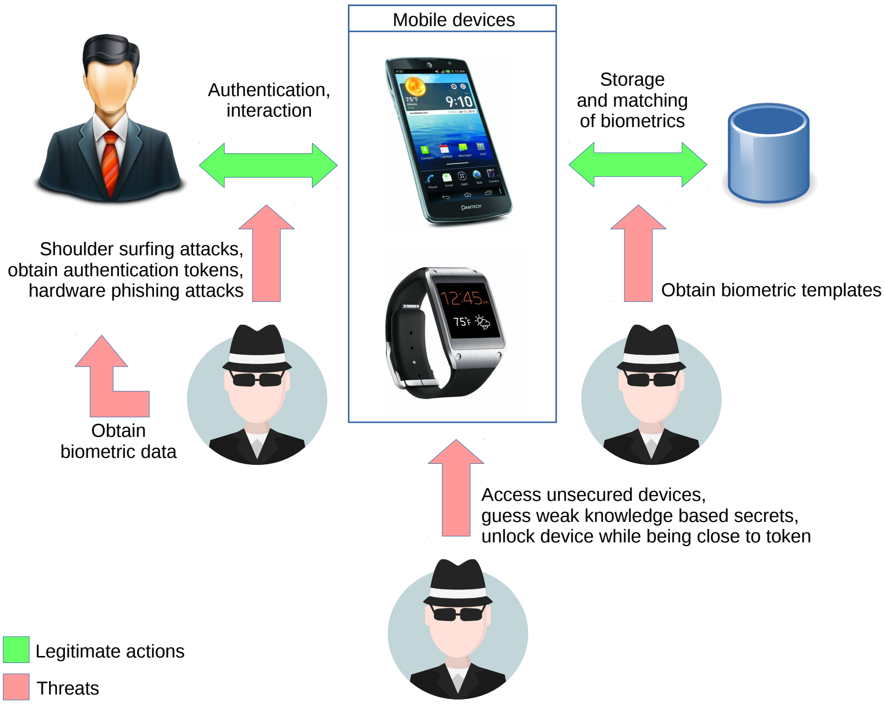

Mobile devices
Authentication, interaction
Storage and matching of biometrics
Shoulder surfing attacks, obtain authentication tokens, hardware phishing attacks
Obtain biometric templates
Obtain biometric data
Access unsecured devices, guess weak knowledge based secrets, unlock device while being close to token
🟩 Legitimate actions
🟥 Threats

Figure 37: Overview of an attacker's perspective on authentication with mobile devices without using our approach.

Attack vectors thereby include obtaining biometric data both from mobile devices and from other sources (e.g. attackers recording biometric data themselves).

With token-based authentication users have to remember to bring the token for authentication. Different authentication systems might require users to carry different tokens. Token-based authentication can also take additional time to locate and present the token to the mobile device. Further, acquisition of tokens is usually connected to additional costs. From a threat model perspective too obtrusive or too costly token-based authentication approaches might again cause users to not use them, leading to unsecured devices. Further, tokens could be accessed or stolen by attackers, potentially together with the corresponding mobile device. In addition, if tokens unlock mobile devices based on proximity, attackers might access a mobile device e.g. behind the back of its owner without being noticed, or unlock it and leave the scene.

In contrast to user-to-device authentication, device-to-user authentication is virtually not used with mobile devices. From a threat model perspective this facilitates hardware phishing attacks in which attackers replace the mobile device with a phishing device. When the unsuspecting user authenticates the authentication information can be relayed to the attackers who can thereby obtain access to the user's device which they previously got under their control.

8.1.2 *Room for Improvement with Mobile Authentication*

The discussed threats emphasize existing issues with current mobile device authentication and illustrate that there is room for improvements regarding certain aspects of mobile authentication. Progress in the corresponding areas would lead to a reduction of attack vectors and improvement of the threat model. From an attackers point of view improvements that would impede the mentioned threats can be summarized as follows:

DIVERSE OPTIONS FOR AUTHENTICATION: the more options for authentication are available the higher the chance that a certain option is unobtrusive in the user's current situation. This would facilitate higher adoption rates of mobile authentication. Advantages in both the combination of different authentication approaches and the diversity of authentication approaches themselves could thereby lead to a reduced threat model.

PROTECTION OF BIOMETRICS: when biometrics are used with mobile authentication their templates need to be protected accordingly. Generic approaches to protecting biometrics could thereby facilitate the protection of different biometrics. As some modern mobile devices are already shipped with SCs these can be used for the purpose of storing and matching biometrics.

DEVICE-TO-USER AUTHENTICATION: first steps towards device-to-user authentication with mobile devices would facilitate the protection of mobile users from hardware phishing attacks. Similar to user-to-device authentication these approaches need to aim for being unobtrusive.

8.2 HOW OUR APPROACH IMPEDES ATTACKS

Our approach fills some of the previously discussed gaps that remain for improving mobile authentication. It impedes some of the corresponding threats and attacks in different ways, thereby contributes to improving the overall threat model of unauthorized physical third party access to mobile devices. We now shortly review how our approach impedes the corresponding threats.

THREAT: ACCESS UNSECURED DEVICES We provide alternative authentication options to mobile authentication like ShakeUnlock and our generic biometric MOC authentication. Those can be used without limitations alongside existing authentication approaches. The more such alternative options for mobile authentication are available the more likely one option suits the user's current situation, leading to overall reduced obtrusiveness. For example, with ShakeUnlock users are able to perform authentication single-handedly without being required to look at the device screen. With an authentication time around 2 s, not bearing any additional cognitive load (users do not need to carry a separate token device as the token is their wrist watch) we argue that ShakeUnlock is a feasible option for authentication on mobile devices and in certain situations less obtrusive than other options. The same applies to our approach to generic biometrics MOC authentication. It does not bear additional cognitive load on users, it protects used biometrics using MOC techniques, and operations involving SCs take around 1-2 s. As our approach is applicable to different biometrics its authentication can suit different situations (such as gait authentication while walking or voice authentication while being on the phone). Our approach thereby contributes to making authentication unobtrusive in different situations, thereby reducing the threat of mobile devices being unsecured.

THREAT: GUESS WEAK KNOWLEDGE BASED SECRETS, SHOULDER SURFING ATTACKS ShakeUnlock and our generic biometric MOC authentication approach are resistant to both users choosing weak secrets and shoulder surfing. This is because both are not utilizing knowledge based authentication at all. Therefore, this threat can be seen as non-existent/addressed with our approach. Still, we want to point out that our approaches are non-perfect in terms of error rates. In situations where our approach is unsuitable, knowledge based authentication could be used as backup strategy. Thereby, one benefit of our approach could be seen as reducing the number of times users are required to use more obtrusive and potentially less secure knowledge based authentication.

THREAT: OBTAIN TOKEN, UNLOCK DEVICE WHILE BEING CLOSE TO TOKEN With our generic biometric MOC authentication, no tokens are involved. With ShakeUnlock, obtaining the token device is arguably more difficult than with classic authentication tokens. Firstly, we do not use a dedicated token device that users might easily forget or lose (which would be explicitly used for authentication, therefore not fulfilling any other purpose). In case of the token device being a wrist watch, it is strapped to the user's wrist. As many people are used to carrying wrist watches, they would implicitly also carry the token device without any additional effort. Further, for obtaining the token, attackers would have to remove the token device from the wrist of the user without the user noticing. Secondly, in contrast to other tokens, our token device features a locking mechanism itself and could easily lock itself when detached from the wrist. To do so it could use e.g. a switch in the latch that locks the device if the latch is opened or embedded sensors that monitor the user's liveness and lock the device when it is removed from the wrist, as no liveness signal can be detected in this case. Therefore, obtaining an unlocked token device is difficult for attackers. Further, ShakeUnlock does rely on proximity of token and mobile device to perform the unlock but uses related acceleration on both devices. Consequently, attackers cannot unlock a mobile device they brought under their control just because of being near to the token device (e.g. behind the user' back), thereby impeding the corresponding threat.

THREAT: OBTAIN BIOMETRIC TEMPLATES Our generic MOC authentication approach is a step towards protecting arbitrary biometrics on SCs using MOC techniques. For attackers, MOC authentication raises the effort required to obtain the involved biometric templates. Attackers are thereby required to run malware on mobile devices instead of being able to read templates from the device storage[1]. Further, as templates stored on the SC with MOC intentionally never leave it, the timing of attacks is limited to when the legitimate user enrolls or authenticates. Our approach thereby is a first step towards protecting arbitrary biometrics using MOC techniques. By being generically applicable to different biometrics and not requiring retraining the model to enroll individual users, it can thereby facilitate the transition of further biometrics to MOC approaches, thereby impeding the threat of attackers obtaining users' biometrics.

THREAT: HARDWARE PHISHING ATTACKS Our approach towards device-to-user authentication ensures users that they are interacting with the correct device and enables them to recognize when interacting with a hardware phishing device instead. Hence our approach impedes hardware phishing attacks being carried out successfully. Our approach should thereby be seen only as a first step towards mobile device-to-user authentication – with the possibility of future research further investigating this topic.

8.3 REMAINING AND NEWLY ARISEN THREATS

While our approach does address threats in authentication with mobile devices it does not cover all potential threats. Some threats have been left out or have been shortened on purpose, others have newly arisen (Fig. 38).

Within the scope of our approach we did not consider strong attackers in detail that are able to e.g. run malware on mobile devices. Such malware could be able to e.g. monitor or manipulate the device memory, sensor values, or the device storage. Malware thereby needs to be considered as a different class of problem: with the mentioned abilities it would be able to undermine the confidentiality and

[1] Attacks on SCs are outside the scope of our work.

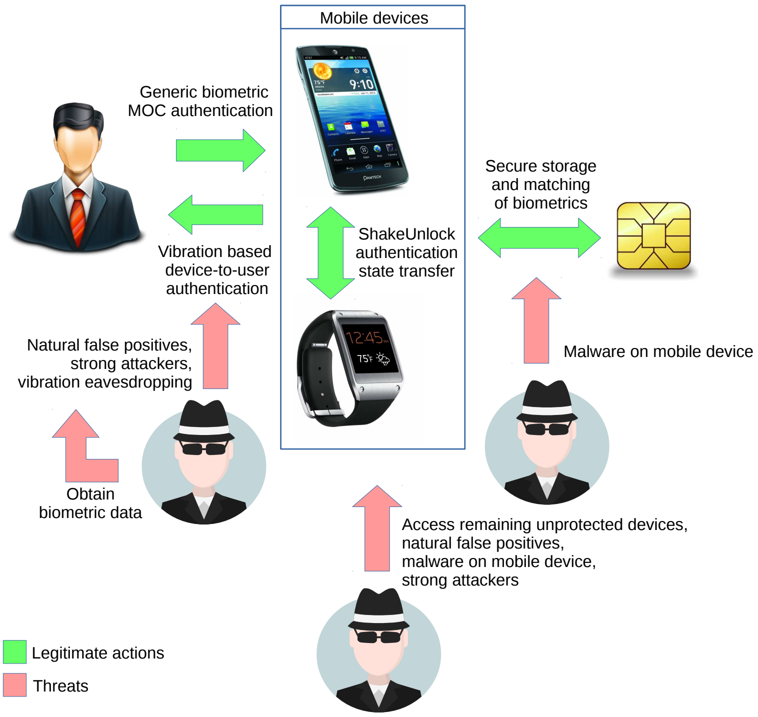

Mobile devices
Generic biometric MOC authentication
Vibration based device-to-user authentication
Secure storage and matching of biometrics
ShakeUnlock authentication state transfer
Natural false positives, strong attackers, vibration eavesdropping
Malware on mobile device
Obtain biometric data
Access remaining unprotected devices, natural false positives, malware on mobile device, strong attackers
🟩 Legitimate actions
🟥 Threats

Figure 38: Overview of threats that remain with applying our approach.

integrity of all information on mobile devices that is not stored or processed within special hardware and completely without influence from the malware. We therefore have declared malware to be out of scope for our approach. We address the problem of mobile authentication under the assumption that there is no malware on devices able to eavesdrop or manipulate information processed internally. Therefore, even when using our approach, a mobile device with malware needs to be considered compromised, therefore incapable of providing means for secure authentication. Countermeasures to malware on mobile devices are an important part of mobile device security that is left to related and future research – however, with our approach, this threat remains. Further, the considered attackers do not use strongly personalized information about their targets to derive information that can be used to circumvent our approaches to mobile authentication. Recent research [239] has suggested and demonstrated an evaluation of biometrics using strong attackers. As such evaluations are strongly dependent on i.e. the used biometrics we leave these aspects for future work. In addition, attackers could use other means than mobile devices to obtain e.g. biometric data about users (for example, attackers recording biometric data themselves). As our approach focuses on mobile devices, such threats are outside the scope and not addressed by our work.

Our approach does not provide perfect user-to-device authentication security results. However, our goal was to provide first steps towards new and additional authentication options that enable unobtrusive authentication while raising the bar for physical access to mobile devices or disclosure of biometrics for attackers. Therefore, there remains room for improvement of authentication accuracies, i.e. using different hardware and more sensors. While perfect authentication security has not been the goal of our work, coincidental authentication errors are a threat that has newly arisen with our work. In terms of device-to-user authentication, another newly arisen threat is the potential for attackers eavesdropping vibration patterns. While eavesdropping vibration is arguably more difficult than eavesdropping authentication information using acoustic or visual signal there is a possibility of attackers e.g. recording and analyzing vibration noise to obtain the device-to-user authentication secret. Investigation of the feasibility of performing such attacks as well as possible countermeasures is left for future work.

Threats that remain with using our approach include too high authentication effort when using a multitude of mobile devices and authentication being impossible to devices with different or no user interfaces. With the first, when users are using a large amount of mobile devices even little effort for authentication might become too much. Users might therefore choose to not protect these devices. While our approach aims to be unobtrusive, one cannot be sure that the required authentication effort (e.g. using biometrics or shaking devices) is sufficiently small when used with the multitude of mobile devices of mobile users in the near and far future. Hence, while we provided means to reduce the amount of unprotected devices some devices might still remain unprotected due to authentication being too obtrusive. In addition, not all types of mobile devices can be protected using our approach. While most devices are equipped with cameras, acceleration sensor, and vibrators (which is sufficient to use all aspects of our approach), devices without these sensors lack the means of employing our approach. Therefore, while the threat of attackers accessing unprotected devices has been reduced, certain devices will remain unprotected, hence parts of this threat remain.

To summarize, our approach is able to address the majority of the previously discussed threats or to provide an improvement to them (Fig. 37), thereby impeding potential attacks and changing an attacker's perspective on mobile authentication. However, other threats remain or have newly arisen with (Fig. 38). We argue that the improvements outweight the newly arisen threats, hence that our approach overall improves the threat model of mobile authentication. We further emphasize that our approach and its constituent parts needs to be seen as first step towards the corresponding directions of providing unobtrusive ways of protecting mobile devices from unauthorized physical access of third parties.

9 CONCLUSION AND OUTLOOK

9.1 SUMMARY

The main objective of this thesis was to investigate additional, alternative, and unobtrusive approaches to authentication with users in the mobile environment in order to protect mobile devices from unauthorized physical access of third parties. The context of this objective, the corresponding research questions, as well as the contributions made by this dissertation within this context have been summarized in Cha. 1. Modern mobile environments with their comprehensive access to diverse data and the applicability of authentication to such environments have been highlighted in Cha. 2. This includes the different types of data mobile devices have access to, the possible impact of this data being disclosed to unauthorized third parties, as well as the applicability of authentication to protect data on mobile devices from unauthorized physical access. Classic authentication like PINs or passwords thereby bring significant drawbacks if applied with mobile devices. Approaches to perform authentication unobtrusively with mobile devices have been discussed in Cha. 3. These comprise diverse knowledge, biometrics, and token-based approaches with respect to the mobile environment and unobtrusiveness, but also approaches incorporating multiple authentication modalities, as well as approaches to let devices authenticate to users. Thereby, while certain previous work has tried to achieve unobtrusiveness with mobile authentication, we conclude that there is still a need for additional approaches that provide for broader possibilities and applicability in diverse situations. This includes both mobile user-to-device authentication as well as device-to-user authentication, whereupon the latter has received little attention in literature.

We presented our approach to unobtrusive bilateral mobile authentication with biometrics and mobile device motion in Cha. 4. Our approach consists of three interconnected parts, namely mobile, generic, and biometric MOC authentication, the transition of authentication states between mobile devices using conjoint shaking, and device-to-user authentication using vibration patterns. With our generic, mobile biometric MOC authentication (Cha. 5) we investigated the simplification of biometric features and offline computed machine learning models for biometric authentication to make their usage on SCs feasible. We obtain one authentication model per biometrics that performs matching of biometric samples on SCs and does not require retraining for enrolling users. By simplifying features and models we further achieve low storage requirements for both models and biometric templates. With ShakeUnlock (Cha. 6) we investigated the transition of authentication states between mobile devices by conjointly shaking them. ShakeUnlock thereby represents a token-based mobile device authentication approach that uses one mobile device to which authentication has already been performed as token and another mobile devices as the target to perform authentication to. Shaking both devices conjointly serves as the trigger mechanism for a transition of the authentication state from one device to another. As shaking is difficult to forge for attackers who only have one device under their control, this further ensures that the transition can only be triggered in case both devices are actually held in the same hand. With our device-to-user authentication (Cha. 7) we investigated using vibration patterns of mobile devices to communicate authentication information from devices to users. We encode a preshared authentication secret with device vibrations and communicate it to users holding the mobile device in their hand. While this is a first step towards the area of mobile device-to-user authentication, this area still remains open for further and novel approaches due to having received little attention in literature in the past altogether.

9.2 CONTRIBUTIONS

This work contributes to unobtrusive mobile authentication with an approach consisting of three interconnected parts. The main contribution can thereby be summarized as follows:

- *The generic protection of biometrics used to perform unobtrusive authentication on mobile devices using SCs:* the novel contribution of our approach comprises of a) combining offline machine learning with simplification of features and models to achieve their employability on computationally restricted SCs with MOC technologies, b) the computed model not requiring retraining for enrolling new users, and c) the approach being generic, that is it being applicable to different biometrics alike.

- *The novel transfer of authentication states between mobile devices in order to perform authentication/unlock them:* our approach uses conjoint shaking of mobile devices in a novel context, namely the transfer of the authentication state from one already unlocked device to another still locked one in order to unlock it. Thereby, this represents a novel mobile and token-based authentication approach that does not impose cognitive load on users and is designed to be applicable in situations where other authentication approaches are more cumbersome to use.

- *Mobile devices authenticating to their users with vibration codes:* our novel approach uses short vibration patterns to communicate an authentication secret to users, e.g. in parallel to them performing user-to-device authentication. Our approach thereby is to be seen as a first step towards mobile device-to-user authentication which has received little attention is past literature.

From the results of the evaluations conducted and the corresponding findings we are able to answer the research questions stated at the beginning of this dissertation (Sec. 1.1):

How can authentication that is employed to protect data on mobile devices from unauthorized physical access of third parties suit the large variety of situations in which authentication might be required?
The previous approaches towards unobtrusive mobile authentication (Cha. 3) point out a) that there are diverse authentication approaches that could be employed with mobile devices and b) that those are often only unobtrusively applicable in certain situations while being obtrusive in others. Further investigating additional, diverse, and alternative ways to mobile authentication, such as with our work (Cha. 4) will likely result in additional approaches becoming available – which can be applied unobtrusively in other situations. The combination of such diverse approaches, e.g. using authentication frameworks like CORMORANT, results in more options and choices being available for mobile authentication. This can result in users being able to choose the best suited authentication in different situations or authentication being performed implicitly and transparently altogether, and thereby result in mobile device authentication becoming less obtrusive.

How could authentication with multiple mobile devices be used as advantage rather than a drawback?
One way to incorporate multiple devices in unobtrusive mobile authentication is to utilize devices to which authentication has already been performed or which are already unlocked as tokens to perform authentication to further devices. In comparison to requiring users to authenticate to each device individually this would result in less often performing classic authentication, thereby can reduce the overall obtrusiveness. Using one device as token its authentication state (e.g. "unlocked") could be transferred to other devices to perform authentication and/or unlock them, as illustrated in our approach with ShakeUnlock (Cha. 6). As with all token-based authentication approaches, an important aspect of such authentication state transfers is to determine when it is secure to be performed. This is necessary to prevent attackers from easily unlocking any mobile devices they got under their control (without being in control of a corresponding token device). Our approach addresses this by requiring both devices to be held in the same hand to trigger an authentication state transfer.

How to protect biometrics used for authentication on mobile devices from disclosure? How to apply such protection to multiple biometrics in order to

aid secure usage of different biometrics on mobile devices in the future?
One way to protect different types of biometrics used for authentication on mobile devices from disclosure is with employing SCs that increasingly become available to current mobile devices. The algorithms employed on SCs should be generically applicable to different biometrics and need to be computable on SCs. Our approach to generic biometric MOC authentication would be one such example (Cha. 5). By simplifying features and authentication models, their usage on SCs becomes possible, both in terms of available memory and storage as well as in sufficiently short computation time. Further, for authentication with modern mobile devices, offline computing one authentication model per biometrics has two significant advantages. Firstly, the model is universally applicable to different participants. It does neither require (re)training to enroll new users nor shipping training data on mobile devices as a prerequisite to perform (re)training, thereby requiring less time and occupying less space on mobile devices. Secondly, as the approach is generic, different models can be computed for different biometrics using the same techniques. Consequently, such generic approaches could aid the transition of further biometrics used with authentication on mobile devices to using MOC techniques.

How can mobile users be protected from hardware phishing attacks, that is them being deceived into unwittingly revealing sensitive information to identically looking but malicious phishing devices?
To protect mobile users from hardware phishing attacks mobile device-to-user authentication can be employed (Sec. 3.6). Thereby, devices authenticate to their users. While different ways of communicating authentication information from devices to users are possible, those are differently difficult to eavesdrop for attackers. This is why we do not employ e.g. a visual or audio but vibration code for this purpose in our approach (Cha. 7). While device-to-user authentication allows for further investigating diverse approaches, those should – similarly to user-to-device authentication – be designed with their obtrusiveness in mind. This includes the cognitive load imposed on users as well as the additional time required to perform authentication. For example, one way of reducing the additional time required for performing device-to-user authentication is to perform it in parallel to users authenticating to their devices.

9.3 CRITICAL EVALUATION AND OUTLOOK FOR FUTURE WORK

While our work has made substantial contributions to unobtrusive mobile authentication it on purpose shortens or leaves out certain aspects which are consequently open for further investigation in the future.

For protecting biometrics used for authentication on mobile devices one aspect left for future work is the protection of information outside secure hardware. While MOC approaches prevent attackers from gaining access to templates stored on the SC or to templates while they are matched they do not secure the whole processing chain. Attackers could obtain access to biometric information while it is outside secure hardware. This applies to both the usage of secure hardware such as SCs and to algorithmic approaches such as biometric template protection. For example, attackers could obtain biometric information by gaining access to sensors or to any preprocessing that is done outside SCs or without biometric template protection. To prevent disclosure on this way additional measures need to be taken. Such could include combining SCs with a trusted execution environment (TEE) that secures the processing chain from the sensor to the SC or biometric template protection. This could also be achieved by integrating the complete processing chain (from sensing biometric information to yielding an authentication decision) into secure hardware, e.g. into a system-on-card (SOC) approach. Alternatively, biometric template protection algorithms could as well be combined with TEEs to protect biometrics information from sensors to yielding an authentication decision. Besides designing suitable and generic MOC approaches, future challenges with protecting biometric information include the design of preprocessing and feature derivation approaches that can be included in TEEs and/or SOC approaches. Theoretically, preprocessing could also be included in biometric template protection algorithms – but in order to secure disclosure of biometric information directly from sensors non-algorithmic approaches are necessarily required.

Another aspect left open for future work is the meaningful combination of diverse authentication modalities. This essentially is the core functionality of authentication framework like CORMORANT which is in the focus of a separate PhD thesis and for which work is currently ongoing. While this present thesis provides for different means of mobile authentication, the combination of their results and the derivation of overall authentication information is left for such future work. Challenges future work will need to address include determining useful levels of confidence (that a legitimate user is operating/trying to authenticate to a mobile device) required to perform certain tasks or access certain data on mobile devices. They further include deriving the risk that mobile devices could be operated by potential attackers, e.g. from the device context, and determining when it is necessary or when there is a good point in time to trigger mobile users for explicit authentication.

Investigating other ways of device-to-user authentication is another aspect left open for future work. While user-to-device authentication has thoroughly been investigated in the past decades, device-to-user authentication has received little attention in literature. This is why this area stays open for novel ideas and approaches, such as with using vibrations to communicate authentication information from devices to users. One detail of using such vibrations that has been left open for future work is the investigation of vibration eavesdropping. While it is arguably more difficult to eavesdrop vibration than visual or audio information communicated from devices to users no quantification of the corresponding effort or success rate has been conducted in our work. As participants in our evaluation assumed that they also used hearing to recognize their vibration authentication pattern conducting experiments to quantify this effect would be interesting. Future work could investigate how big the contribution of hearing is when recognizing vibration using a combination of feeling and hearing and how well both work when used individually. It could further investigate how easy it would be for attackers to eavesdrop information communicated by different mobile devices using vibrations when only being able to hear it from certain distance and including e.g. different types and amounts of background noise.

A more general aspect that has partially been left for future work is the inclusion of strong and powerful attackers. While the resistance of our approach has been shown for types of attackers typically used in literature, resistance against strong and powerful attackers can be considerably more difficult. For zero or minimal effort attacks attackers are not required to have comprehensive knowledge about their targets. While this lowers the effort required to perform such attacks, it can also be connected to decreasing chances of attacks being successful. In contrast, attackers having comprehensive knowledge about their targets might enable different forms of attacks altogether. One recent and illustrative example of using strong attackers in evaluations of biometric authentication approaches would include the work of Muaaz and Mayrhofer [239] which use trained actors to copy human gait as good as possible. While they find that in their evaluation both weak and strong attackers are unable to break their approach this might not be the case for other approaches. The main challenge of evaluating diverse authentication approaches with strong attackers is modeling such attackers in the first place, as the possibilities and ways to perform attacks rise drastically with the power and resources attackers have access to. Investigating how to model strong attackers for diverse authentication approaches and evaluating those approaches using strong attackers thereby is an important point left open for future work.

To summarize: in the short term, future work on unobtrusive mobile authentication could further investigate additional and alternative authentication approaches. While each individual approach might only be unobtrusively applicable in certain situations, the combination of such approaches could provide for increasingly unobtrusive authentication on mobile devices in the future. Especially implicit, continuous, and in certain situations completely transparent authentication could contribute to this. In the medium term, future work on unobtrusive mobile authentication could investigate different ways to incorporate the increasing amount of mobile devices users own or have access to, similarly to e.g. ShakeUnlock or the cross-device aspect [147, 148] of CORMORANT, which is still under active development. In the long term, future work on unobtrusive mobile authentication will consequently have to deal with the multitude of devices that might become part of our daily lives, e.g. with currently heavily researched areas such as the Internet-of-Things (IoT) or the area of automotive computing. Another important long-term aspect of mobile authentication will be the protection of users' digital identities. Interactions in a fully digital environment will lead to the digital representation of users becoming of significant importance in the future. Protecting access to this identity – which will necessarily be available in a mobile manner – will be another important aspect of mobile authentication in the future.

BIBLIOGRAPHY

[1] A. F. Abate, M. Nappi, D. Riccio, and G. Sabatino. "2D and 3D face recognition: a survey." In: *Pattern Recognition Letters* 28.14 (Oct. 2007), pp. 1885–1906.

[2] S. N. Abdulkader, A. Atia, and M.-S. M. Mostafa. "Authentication systems: principles and threats." In: *Computer and Information Science* 8.3 (2015).

[3] A. Adams and M. A. Sasse. "Users are not the enemy." In: *Communications of the ACM* 42.12 (Dec. 1999), pp. 40–46.

[4] J. Adkins, G. Flaspohler, and P. Dutta. "Ving: bootstrapping the desktop area network with a vibratory ping." In: *The 2nd ACM Workshop on Hot Topics in Wireless (HotWireless'15)*. Paris, France, Sept. 2015.

[5] A. Adler. "Sample images can be independently restored from face recognition templates." In: *Canadian Conference on Electrical and Computer Engineering 2003 (IEEE CCECE 2003)*. Vol. 2. May 2003, pp. 1163–1166.

[6] F. Ahmad and D. Mohamad. "A review on fingerprint classification techniques." In: *International Conference on Computer Technology and Development 2009*. Vol. 2. Nov. 2009, pp. 411–415.

[7] I. Ahmed, Y. Ye, S. Bhattacharya, N. Asokan, G. Jacucci, P. Nurmi, and S. Tarkoma. "Checksum gestures: continuous gestures as an out-of-band channel for secure pairing." In: *Proc. UbiComp 2015*. Osaka, Japan: ACM, 2015, pp. 391–401.

[8] K. Airowaily and M. Alrubaian. "Oily residuals security threat on smart phones." In: *First International Conference on Robot, Vision and Signal Processing*. Nov. 2011, pp. 300–302.

[9] A. Al Abdulwahid, N. Clarke, I. Stengel, S. Furnell, and C. Reich. "Continuous and transparent multimodal authentication: reviewing the state of the art." In: *Cluster Computing* 19.1 (2016), pp. 455–474.

[10] S. Alotaibi, S. Furnell, and N. Clarke. "Transparent authentication systems for mobile device security: a review." In: *Proc. 10th International Conference for Internet Technology and Secured Transactions (ICITST 2015)*. Dec. 2015, pp. 406–413.

[11] K. Altun, B. Barshan, and O. Tunçel. "Comparative study on classifying human activities with miniature inertial and magnetic sensors." In: *Pattern Recognition* 43.10 (Oct. 2010), pp. 3605–3620.

[12] F. H. Álvarez, L. H. Encinas, and C. Sanchez-Avila. "Biometric fuzzy extractor scheme for iris templates." In: *Security and Management*. Ed. by H. R. Arabnia and K. Daimi. CSREA Press, 2009, pp. 563–569.

[13] S. Antifakos, B. Schiele, and L. E. Holmquist. "Grouping mechanisms for smart objects based on implicit interaction and context proximity." In: *Proc. UbiComp 2003 Interactive Posters*. 2003, pp. 207–208.

[14] A. Arakala, J. Jeffers, and K. Horadam. "Fuzzy extractors for minutiae-based fingerprint authentication." In: *Advances in Biometrics*. Ed. by S.-W. Lee and S. Li. Vol. 4642. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, pp. 760–769.

[15] H. Assal, S. Hurtado, A. Imran, and S. Chiasson. "What's the deal with privacy apps?: a comprehensive exploration of user perception and usability." In: *Proc. 14th International Conference on Mobile and Ubiquitous Multimedia (MUM 2015)*. Linz, Austria: ACM, 2015, pp. 25–36.

[16] P. J. Attwell, S. F. Cooke, and C. H. Yeo. "Cerebellar function in consolidation of a motor memory." In: *Neuron* 34.6 (2002), pp. 1011–1020.

[17] A. J. Aviv, K. Gibson, E. Mossop, M. Blaze, and J. M. Smith. "Smudge attacks on smartphone touch screens." In: *Proc. of the 4th USENIX conference on offensive technologies*. Washington, DC, 2010, pp. 1–7.

[18] M. Bächlin, J. Schumm, D. Roggen, and G. Töster. "Quantifying gait similarity: user authentication and real-world challenge." In: *3rd International Conference on Advances in Biometrics (ICB 2009)*. Ed. by M. Tistarelli and M. S. Nixon. Berlin, Heidelberg: Springer Berlin Heidelberg, June 2009, pp. 1040–1049.

[19] M. Baloul, E. Cherrier, and C. Rosenberger. “Challenge-based speaker recognition for mobile authentication.” In: *Proc. International Conference of the Biometrics Special Interest Group (BIOSIG 2012)*. 2012, pp. 1–7.

[20] L. Bao and S. T. c. n. Intille. “Activity recognition from user-annotated acceleration data.” In: *Pervasive Computing*. Ed. by A. Ferscha and F. Mattern. Vol. 3001. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2004, pp. 1–17.

[21] P. Bao, J. Pierce, S. Whittaker, and S. Zhai. “Smart phone use by non-mobile business users.” In: *Proc. 13th International Conference on Human Computer Interaction with Mobile Devices and Services*. MobileHCI 2011. Stockholm, Sweden: ACM, 2011, pp. 445–454.

[22] S. Barra, A. Casanova, F. Narducci, and S. Ricciardi. “Ubiquitous iris recognition by means of mobile devices.” In: *Pattern Recognition Letters* 57 (2015), pp. 66–73.

[23] T. Beauvisage. “Computer usage in daily life.” In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI 2009. Boston, MA, USA: ACM, 2009, pp. 575–584.

[24] H. Beigi. *Fundamentals of speaker recognition*. Springer Science & Business Media, 2011.

[25] N. Belgacem, A. Ali, R. Fournier, and F. Bereksi-Reguig. “ECG based human authentication using wavelets and random forests.” In: *International Journal on Cryptography and Information Security (IJCIS)* 2.2 (2012), pp. 1–11.

[26] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. “Eigenfaces vs. fisherfaces: recognition using class specific linear projection.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19.7 (July 1997), pp. 711–720.

[27] N. Ben-Asher, N. Kirschnick, H. Sieger, J. Meyer, A. Ben-Oved, and S. Möller. “On the need for different security methods on mobile phones.” In: *Proc. 13th International Conference on Human Computer Interaction with Mobile Devices and Services*. MobileHCI 2011. Stockholm, Sweden: ACM, 2011, pp. 465–473.

[28] H. Ben-Pazi, H. Bergman, J. A. Goldberg, N. Giladi, D. Hansel, A. Reches, and E. S. Simon. “Synchrony of rest tremor in multiple limbs in parkinson’s disease: evidence for multiple oscillators.” In: *Journal of Neural Transmission* 108.3 (2001), pp. 287–296.

[29] H.-G. Beyer. *The Theory of Evolution Strategies*. New York, NY, USA: Springer-Verlag New York, Inc., 2001.

[30] D. Bichler, G. Stromberg, M. Huemer, and M. Löw. “Key generation based on acceleration data of shaking processes.” In: *Proc. 9th International Conference on Ubiquitous Computing*. UbiComp 2007. Innsbruck, Austria: Springer-Verlag, 2007, pp. 304–317.

[31] R. Biddle, S. Chiasson, and P. C. van Oorschot. “Graphical passwords: learning from the first twelve years.” In: *ACM Compututing Surveys (CSUR)* 44.4 (Sept. 2012), 19:1–19:41.

[32] S. Bistarelli, F. Santini, and A. Vaccarelli. “An asymmetric fingerprint matching algorithm for Java Card.” In: *Pattern Analysis and Applications* 9.4 (2006), pp. 359–376.

[33] R. Blanco-Gonzalo, O. Miguel-Hurtado, A. Mendaza-Ormaza, and R. Sanchez-Reillo. “Handwritten signature recognition in mobile scenarios: performance evaluation.” In: *IEEE International Carnahan Conference on Security Technology (ICCST 2012)*. Oct. 2012, pp. 174–179.

[34] G. Blonder. *Graphical password*. Sept. 1996.

[35] H. Bojinov and D. Boneh. “Mobile token-based authentication on a budget.” In: *Proc. 12th Workshop on Mobile Computing Systems and Applications*. HotMobile 2011. Phoenix, Arizona: ACM, 2011, pp. 14–19.

[36] J. Bonnau, C. Herley, P. C. van Oorschot, and F. Stajano. “The quest to replace passwords: a framework for comparative evaluation of web authentication schemes.” In: *Proc. 2012 IEEE Symposium on Security and Privacy (SOUPS 2012)*. Washington, DC, USA: IEEE Computer Society, 2012, pp. 553–567.

[37] J. Bonneau. “The science of guessing: analyzing an anonymized corpus of 70 million passwords.” In: *IEEE Symposium on Security and Privacy (SP 2012)*. 2012, pp. 538–552.

[38] T. Bourlai, K. Messer, and J. Kittler. “Face verification system architecture using smart cards.” In: *Proc. ICPR 2004*. Vol. 1. Aug. 2004, pp. 793–796.

[39] P. Bours and R. Shrestha. “Eigensteps: a giant leap for gait recognition.” In: *Security and Communication Networks (IWSCN) 2010*. May 2010, pp. 1–6.

[40] C. Bouten, K. Koekkoek, M. Verduin, R. Kodde, and J. Janssen. “A triaxial accelerometer and portable data processing unit for the assessment of daily physical activity.” In: *IEEE Biomedical Engineering* 44.3 (1997), pp. 136–147.

[41] K. W. Bowyer, K. Chang, and P. Flynn. "A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition." In: *Computer Vision and Image Understanding* 101.1 (2006), pp. 1–15.

[42] K. W. Bowyer, K. Hollingsworth, and P. J. Flynn. "Image understanding for iris biometrics: a survey." In: *Computer Vision and Image Understanding* 110.2 (May 2008), pp. 281–307.

[43] J. Breebaart, B. Yang, I. Buhan-Dulman, and C. Busch. "Biometric template protection." In: *Datenschutz und Datensicherheit - DuD* 33.5 (2009), pp. 299–304.

[44] J. Bringer, H. Chabanne, D. Le Métayer, and R. Lescuyer. "Privacy by design in practice: reasoning about privacy properties of biometric system architectures." In: *Formal Methods (FM) 2015*. Vol. 9109. Lecture Notes in Computer Science. Springer, 2015, pp. 90–107.

[45] S. Brostoff and M. A. Sasse. "Are passfaces more usable than passwords? a field trial investigation." In: *People and Computers XIV — Usability or Else!: Proc. HCI 2000*. Ed. by S. McDonald, Y. Waern, and G. Cockton. London: Springer London, 2000, pp. 405–424.

[46] I. Buhan, B. Boom, J. Doumen, P. H. Hartel, and R. N. J. Veldhuis. "Secure pairing with biometrics." In: *International Journal of Security and Networks* 4.1/2 (Feb. 2009), pp. 27–42.

[47] I. Buhan, J. Doumen, P. Hartel, and R. Veldhuis. "Fuzzy extractors for continuous distributions." In: *Proc. 2nd ACM Symposium on Information, Computer and Communications Security*. ASIACCS '07. Singapore: ACM, 2007, pp. 353–355.

[48] I. Buhan, J. Doumen, P. Hartel, and R. Veldhuis. "Secure ad-hoc pairing with biometrics: safe." In: *First International Workshop on Security for Spontaneous Interaction*. Innsbruck, Austria: Ubicomp 2007 Workshop Proceedings, Sept. 2007, pp. 450–456.

[49] I. Buhan, J. Doumen, P. H. Hartel, and R. N. Veldhuis. *Constructing practical Fuzzy Extractors using QIM*. Tech. rep. Twente, Netherlands: Faculty of Electrical Engineering, Mathematics & Computer Science, University of Twente, June 2007.

[50] A. Buriro, B. Crispo, and Y. Zhauniarovich. "Please hold on: unobtrusive user authentication using smartphone's built-in sensors." In: *2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA)*. Feb. 2017, pp. 1–8.

[51] A. Buriro, B. Crispo, F. Del Frari, J. Klardie, and K. Wrona. "Itsme: multi-modal and unobtrusive behavioural user authentication for smartphones." In: *Technology and Practice of Passwords: 9th International Conference, PASSWORDS 2015, Cambridge, UK, December 7–9, 2015, Proceedings*. Ed. by F. Stajano, S. F. Mjølsnes, G. Jenkinson, and P. Thorsheim. Springer International Publishing, 2016, pp. 45–61.

[52] W. M. Campbell, D. E. Sturim, and D. A. Reynolds. "Support vector machines using GMM supervectors for speaker verification." In: *IEEE Signal Processing Letters* 13.5 (May 2006), pp. 308–311.

[53] K. Cao and A. Jain. "Learning fingerprint reconstruction: from minutiae to image." In: *IEEE Information Forensics and Security* 10.1 (Jan. 2015), pp. 104–117.

[54] R. Cappelli and D. Maio. "The state of the art in fingerprint classification." In: *Automatic Fingerprint Recognition Systems*. Ed. by N. Ratha and R. Bolle. New York, NY: Springer New York, 2004, pp. 183–205.

[55] R. Cappelli, D. Maio, A. Lumini, and D. Maltoni. "Fingerprint image reconstruction from standard templates." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.9 (Sept. 2007), pp. 1489–1503.

[56] C. Castelluccia and P. Mutaf. "Shake them up!: a movement-based pairing protocol for CPU-constrained devices." In: *Proc. of the 3rd International Conference on Mobile Systems, Applications, and Services*. MobiSys '05. Seattle, Washington: ACM, 2005, pp. 51–64.

[57] A. Cavoukian and A. Stoianov. "Biometric encryption." In: *Encyclopedia of Biometrics*. Springer, 2009.

[58] T.-Y. Chang, C.-J. Tsai, and J.-H. Lin. "A graphical-based password keystroke dynamic authentication system for touch screen handheld mobile devices." In: *Journal of Systems and Software* 85.5 (2012), pp. 1157–1165.

[59] J. Chapran. "Biometric writer identification: feature analysis and classification." In: *International Journal of Pattern Recognition and Artificial Intelligence* 20.04 (2006), pp. 483–503.

[60] Y. Chen and M. Sinclair. "Tangible security for mobile devices." In: *Proc. 5th Annual International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services*. Mobiquitous '08. Dublin, Ireland: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, 19:1–19:4.

[61] H.-Y. Chiang and S. Chiasson. "Improving user authentication on mobile devices: a touchscreen graphical password." In: *Proc. 15th International Conference on Human-computer Interaction with Mobile Devices and Services*. MobileHCI '13. Munich, Germany: ACM, 2013, pp. 251–260.

[62] S. Chiasson, R. Biddle, and P. C. van Oorschot. "A second look at the usability of click-based graphical passwords." In: *Proc. 3rd Symposium on Usable Privacy and Security*. SOUPS '07. Pittsburgh, Pennsylvania, USA: ACM, 2007, pp. 1–12.

[63] S. Chiasson, A. Forget, R. Biddle, and P. C. van Oorschot. "Influencing users towards better passwords: persuasive cued click-points." In: *Proc. 2nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction*. Vol. 1. BCS-HCI '08. Liverpool, United Kingdom: British Computer Society, 2008, pp. 121–130.

[64] S. Chiasson, A. Forget, R. Biddle, and P. C. van Oorschot. "User interface design affects security: patterns in click-based graphical passwords." In: *International Journal of Information Security* 8.6 (2009), p. 387.

[65] S. Chiasson, P. C. van Oorschot, and R. Biddle. "Graphical password authentication using cued click points." In: *12th European Symposium On Research In Computer Security (ESORICS 2007)*. Ed. by J. Biskup and J. López. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 359–374.

[66] W.-Y. Choi, D. Ahn, S. B. Pan, K. I. Chung, Y. Chung, and S.-H. Chung. "SVM-based speaker verification system for match-on-card and its hardware implementation." In: *E-lectronics and Telecommunications Research Institute Journal (ETRI)* 28.3 (June 2006), pp. 320–328.

[67] M. K. Chong, R. Mayrhofer, and H. Gellersen. "A survey of user interaction for spontaneous device association." In: *ACM Computing Surveys* 47.1 (May 2014), 8:1–8:40.

[68] N. Clarke and S. Furnell. "Authentication of users on mobile telephones – a survey of attitudes and practices." In: *Computers and Security* 24.7 (2005), pp. 519–527.

[69] P. Coli, G. L. Marcialis, and F. Roli. "Vitality detection from fingerprint images: a critical survey." In: *International Conference on Advances in Biometrics (ICB 2007)*. Ed. by S.-W. Lee and S. Z. Li. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 722–731.

[70] C. T. Cornelius and D. F. Kotz. "Recognizing whether sensors are on the same body." In: *Pervasive and Mobile Computing* 8.6 (2012), pp. 822–836.

[71] M. D. Corner and B. D. Noble. "Zero-interaction authentication." In: *Proc. 8th Annual International Conference on Mobile Computing and Networking*. MobiCom '02. Atlanta, Georgia, USA: ACM, 2002, pp. 1–11.

[72] L. F. Cranor and S. Garfinkel. *Security and Usability*. O'Reilly Media, May 2008.

[73] D. Crouse, H. Han, D. Chandra, B. Barbello, and A. K. Jain. "Continuous authentication of mobile user: fusion of face image and inertial measurement unit data." In: *2015 International Conference on Biometrics (ICB)*. May 2015, pp. 135–142.

[74] J. Cukier and W. Liang. *Token-enabled authentication for securing mobile devices*. June 2007.

[75] A. Czajka, P. Strzelczyk, M. Chochowski, and A. Pacut. "Iris recognition with match-on-card." In: *Proc. European Signal Processing Conference (EUSIPCO)*. Poznan, Poland, Sept. 2007, pp. 189–192.

[76] W. Dargie. "Analysis of time and frequency domain features of accelerometer measurements." In: *Proc. of 18th Internatonal Conference on Computer Communications and Networks (ICCCN 2009)*. 2009, pp. 1–6.

[77] W. Dargie and M. Denko. "Analysis of error-agnostic time- and frequency-domain features extracted from measurements of 3D accelerometer sensors." In: *IEEE Systems Journal* 4.1 (2010), pp. 26–33.

[78] I. Daubechies. "Orthonormal bases of compactly supported wavelets II. variations on a theme." In: *SIAM Journal on Mathematical Analysis* 24.2 (1993), pp. 499–519.

[79] J. Daugman. "New methods in iris recognition." In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 37.5 (2007), pp. 1167–1175.

[80] J. Daugman. "How iris recognition works." In: *IEEE Transactions on Circuits Systems and Video Technologies* 14.1 (2004), pp. 21–30.

[81] J. Daugman. “Probing the uniqueness and randomness of IrisCodes: Results from 200 billion iris pair comparisons.” In: *Proc. IEEE* 94.11 (Nov. 2006), pp. 1927–1935.

[82] J. Daugman and C. Downing. “Epigenetic randomness, complexity, and singularity of human iris patterns.” In: *Proc. Royal Society, B, Biological Sciences* 268 (2001), pp. 1737–1740.

[83] C. R. Davis. *IPCec: Securing VPNs*. McGraw-Hill Professional, 2001.

[84] D. Davis, F. Monrose, and M. K. Reiter. “On user choice in graphical password schemes.” In: *Proc. 13th Conference on USENIX Security Symposium - Volume 13*. SSYM’04. San Diego, CA: USENIX Association, 2004, pp. 11–11.

[85] A. De Luca, A. Hang, F. Brudy, C. Lindner, and H. Hussmann. “Touch me once and I know it’s you! implicit authentication based on touch screen patterns.” In: *Proc. 2012 ACM annual conference on Human Factors in Computing Systems*. CHI ’12. New York, NY, USA: ACM, 2012, pp. 987–996.

[86] A. De Luca, E. von Zezschwitz, N. D. H. Nguyen, M.-E. Maurer, E. Rubegni, M. P. Scipioni, and M. Langheinrich. “Back-of-device authentication on smartphones.” In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’13. New York, NY, USA: ACM, 2013, pp. 2389–2398.

[87] K. Delac and M. Grgic. “A survey of biometric recognition methods.” In: *Electronics in Marine, 2004. Proceedings Elmar 2004. 46th International Symposium*. 2004, pp. 184–193.

[88] M. O. Derawi. “Biometric options for mobile phone authentication.” In: *Biometric Technology Today* 2011.9 (2011), pp. 5–7.

[89] A. E. Dirik, N. Memon, and J.-C. Birget. “Modeling user choice in the passpoints graphical password scheme.” In: *Proc. 3rd Symposium on Usable Privacy and Security*. SOUPS ’07. Pittsburgh, Pennsylvania, USA: ACM, 2007, pp. 20–28.

[90] M. Dobes and L. Machala. “UPOL iris image database, 2004.” In: *Available at: http;//www. phoenix. inf. upol. cz/iris* (2013).

[91] A. Dobson. *An Introduction to Generalized Linear Models, Second Edition*. Chapman & Hall/CRC Texts in Statistical Science. Taylor & Francis, 2010.

[92] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith. “Fuzzy extractors: how to generate strong keys from biometrics and other noisy data.” In: *SIAM J. Comput.* 38.1 (Mar. 2008), pp. 97–139.

[93] Y. Dodis, L. Reyzin, and A. Smith. “Fuzzy extractors.” In: *Security with Noisy Data*. Ed. by P. Tuyls, B. Skoric, and T. Kevenaar. Springer London, 2007, pp. 79–99.

[94] P. Dunphy, A. P. Heiner, and N. Asokan. “A closer look at recognition-based graphical passwords on mobile devices.” In: *Proc. Sixth Symposium on Usable Privacy and Security*. SOUPS ’10. Redmond, Washington: ACM, 2010, 3:1–3:12.

[95] P. Dunphy and J. Yan. “Do background images improve "draw a secret"graphical passwords?” In: *Proc. 14th ACM Conference on Computer and Communications Security*. CCS ’07. Alexandria, Virginia, USA: ACM, 2007, pp. 36–47.

[96] M. Engin, S. Demirağ, E. Z. Engin, G. Çelebi, F. Ersan, E. Asena, and Z. Çolakoğlu. “The classification of human tremor signals using artificial neural network.” In: *Expert Systems with Applications* 33.3 (2007), pp. 754–761.

[97] M. J. Er, S. Wu, J. Lu, and H. L. Toh. “Face recognition with radial basis function (RBF) neural networks.” In: *IEEE Transactions on Neural Networks* 13.3 (May 2002), pp. 697–710.

[98] K. M. Everitt, T. Bragin, J. Fogarty, and T. Kohno. “A comprehensive study of frequency, interference, and training of multiple graphical passwords.” In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’09. Boston, MA, USA: ACM, 2009, pp. 889–898.

[99] N. Fatima and T. Zheng. “Short utterance speaker recognition a research agenda.” In: *International Conference on Systems and Informatics (ICSAI) 2012*. 2012, pp. 1746–1750.

[100] T. Feng, X. Zhao, B. Carbunar, and W. Shi. “Continuous mobile authentication using virtual key typing biometrics.” In: *12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*. July 2013, pp. 1547–1552.

[101] P. Fernández Clotet and R. D. Findling. “Mobile wrist vein authentication using SIFT features.” In: *Proc. Eurocast 2017*. Las Palmas, Spain: Springer, Apr. 2017.

[102] R. D. Findling. “Pan Shot Face Unlock: Towards Unlocking Personal Mobile Devices using Stereo Vision and Biometric Face Information from multiple Perspectives.” MA thesis. Softwarepark 11, 4232 Hagenberg/Austria: Department of Mobile Computing, School of Informatics, Communication and Media, University of Applied Sciences Upper Austria, Sept. 2013.

[103] R. D. Findling, M. Hölzl, and R. Mayrhofer. “Mobile gait match-on-card authentication from acceleration data with offline-simplified models.” In: *Proc. MoMM 2016: 14th International Conference on Advances in Mobile Computing and Multimedia*. Singapore: ACM, Nov. 2016, pp. 250–260.

[104] R. D. Findling, M. Hölzl, and R. Mayrhofer. “Mobile match-on-card authentication using offline-simplified models with gait and face biometrics.” In: *IEEE Transactions on Mobile Computing (TMC)* (2017).

[105] R. D. Findling and R. Mayrhofer. “Towards device-to-user authentication: protecting against phishing hardware by ensuring mobile device authenticity using vibration patterns.” In: *14th International Conference on Mobile and Ubiquitous Multimedia (MUM'15)*. ACM, Dec. 2015, pp. 131–136.

[106] R. D. Findling, M. Muaaz, D. Hintze, and R. Mayrhofer. “ShakeUnlock: securely unlock mobile devices by shaking them together.” In: *Proc. MoMM 2014: 12th International Conference on Advances in Mobile Computing and Multimedia*. Kaohsiung, Taiwan: ACM Press, Dec. 2014, pp. 165–174.

[107] R. D. Findling, M. Muaaz, D. Hintze, and R. Mayrhofer. “Shakeunlock: securely transfer authentication states between mobile devices.” In: *IEEE Transactions on Mobile Computing (TMC)* 16.4 (Apr. 2017), pp. 1163–1175.

[108] S. Flügge, H. Scharf, S. Fahl, and M. Smith. “Poster: preliminary investigation of an NFC-unlock mechanism for Android.” In: *SOUPS '13: Proc. Ninth Symposium on Usable Privacy and Security*. Newcastle, United Kingdom: ACM, 2013.

[109] A. Forget, S. Chiasson, and R. Biddle. “Choose your own authentication.” In: *New Security Paradigm Workshop (NSPW)*. ACM. 2015.

[110] M. Freire-Santos, J. Fierrez-Aguilar, and J. Ortega-Garcia. “Cryptographic key generation using handwritten signature.” In: *Biometric Technology for Human Identification III*. Ed. by P. J. Flynn and S. Pankanti. Kissimmee, Orlando, Florida, Apr. 2006.

[111] L. Fridman, S. Weber, R. Greenstadt, and M. Kam. “Active authentication on mobile devices via stylometry, application usage, web browsing, and GPS location.” In: *IEEE Systems Journal* 11.2 (June 2017), pp. 513–521.

[112] K. Fujinami and S. Pirttikangas. “A study on a correlation coefficient to associate an object with its user.” In: *3rd IET International Conference on Intelligent Environments (IE 2007)*. 2007, pp. 288–295.

[113] D. Gafurov, E. Snekkenes, and P. Bours. “Gait authentication and identification using wearable accelerometer sensor.” In: *Automatic Identification Advanced Technologies*. June 2007, pp. 220–225.

[114] D. Gafurov. “Performance and Security Analysis of Gait-based User Authentication.” PhD thesis. Faculty of Mathematics and Natural Sciences at the University of Oslo, 2008.

[115] D. Gafurov and E. Snekkenes. “Gait recognition using wearable motion recording sensors.” In: *EURASIP Advances in Signal Processing* 2009 (Jan. 2009), 7:1–7:16.

[116] M. Galar et al. “A survey of fingerprint classification part i: taxonomies on feature extraction methods and learning models.” In: *Knowledge-Based Systems* 81 (2015), pp. 76–97.

[117] J. Galbally, C. McCool, J. Fierrez, S. Marcel, and J. Ortega-Garcia. “On the vulnerability of face verification systems to hill-climbing attacks.” In: *Pattern Recognition* 43.3 (2010), pp. 1027–1038.

[118] F. Galton. *Finger prints*. Macmillan and Company, 1892.

[119] J. Garcia. *Mobile wireless communications device performing device unlock based upon near field communication (nfc) and related methods*. Dec. 2014.

[120] S. Garcia-Salicetti, C. Beumier, G. Chollet, B. Dorizzi, J. L. l. Jardins, J. Lunter, Y. Ni, and D. Petrovska-Delacrétaz. “BIOMET: a multimodal person authentication database including face, voice, fingerprint, hand and signature modalities.” In: *4th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA 2003)*. Ed. by J. Kittler and M. S. Nixon. Berlin, Heidelberg: Springer, 2003, pp. 845–853.

[121] A. Goh and D. C. L. Ngo. "Computation of cryptographic keys from face biometrics." In: *Communications and Multimedia Security. Advanced Techniques for Network and Data Protection*. Ed. by A. Lioy and D. Mazzocchi. Vol. 2828. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2003, pp. 1–13.

[122] J. Goldberg, J. Hagman, and V. Sazawal. "Doodling our way to better authentication." In: *CHI '02 Extended Abstracts on Human Factors in Computing Systems*. CHI EA '02. Minneapolis, Minnesota, USA: ACM, 2002, pp. 868–869.

[123] A. Goode. "Bring your own finger – how mobile is bringing biometrics to consumers." In: *Biometric Technology Today* 2014.5 (2014), pp. 5–9.

[124] L. Gorman. "Comparing passwords, tokens, and biometrics for user authentication." In: *Proc. IEEE* 91.12 (Dec. 2003), pp. 2021–2040.

[125] M. Govan and T. Buggy. "A computationally efficient fingerprint matching algorithm for implementation on smartcards." In: *Biometrics: Theory, Applications, and Systems (BTAS) 2007*. Sept. 2007, pp. 1–6.

[126] N. S. Govindarajulu and S. Madhvanath. "Password management using doodles." In: *Proc. 9th International Conference on Multimodal Interfaces*. ICMI '07. Nagoya, Aichi, Japan: ACM, 2007, pp. 236–239.

[127] R. Greenstadt, M. Kam, L. Fridman, and P. Brenna. *Mobile Active Authentication via Linguistic Modalities*. Tech. rep. Philadelphia, Pennsylvania: Drexel University, 2015.

[128] E. Grosse and M. Upadhyay. "Authentication at scale." In: *IEEE Security and Privacy* 11 (2013), pp. 15–22.

[129] P. Grother, W. Salamon, C. Watson, M. Indovina, and P. Flanagan. *MINEX II: Performance of Fingerprint Match-on-Card Algorithms Phase II / III Report. NIST Interagency Report 7477 (Rev. I)*. Tech. rep. Information Access Division, National Institute of Standards and Technology (NIST), May 2009.

[130] B. Groza and R. Mayrhofer. "SAPHE: simple accelerometer based wireless pairing with heuristic trees." In: *Proc. of the 10th International Conference on Advances in Mobile Computing & Multimedia*. MoMM '12. Bali, Indonesia: ACM, 2012, pp. 161–168.

[131] M. Hacker, M. Crovella, and L. Reyzin. "Secure Pairing of Mobile Devices." MA thesis. Boston University, May 2012.

[132] M. Hafiz, A. Abdullah, N. Ithnin, and H. Mammi. "Towards identifying usability and security features of graphical password in knowledge based authentication technique." In: *Second Asia International Conference on Modeling Simulation (AICMS 08)*. May 2008, pp. 396–403.

[133] S. Hallsteinsen, I. Jorstad, and D. V. Thanh. "Using the mobile phone as a security token for unified authentication." In: *2007 Second International Conference on Systems and Networks Communications (ICSNC 2007)*. Aug. 2007, pp. 68–68.

[134] M. Hansen, R. Hill, and S. Wimberly. "Detecting covert communication on Android." In: *IEEE 37th Conference on Local Computer Networks (LCN 2012)*. Oct. 2012, pp. 300–303.

[135] M. Harbach, A. De Luca, and S. Egelman. "The anatomy of smartphone unlocking: a field study of Android lock screens." In: *Proc. 2016 CHI Conference on Human Factors in Computing Systems*. CHI '16. Santa Clara, California, USA: ACM, 2016, pp. 4806–4817.

[136] M. Harbach, E. von Zezschwitz, A. Fichtner, A. D. Luca, and M. Smith. "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)*. Menlo Park, CA: USENIX Association, July 2014, pp. 213–230.

[137] H. Hasan and S. Abdul-Kareem. "Fingerprint image enhancement and recognition algorithms: a survey." In: *Neural Computing and Applications* 23.6 (2013), pp. 1605–1610.

[138] M. R. Hasan, M. Jamil, M. G. R. M. S. Rahman, et al. "Speaker identification using mel frequency cepstral coefficients." In: *Variations* 1.4 (2004).

[139] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning: Data Mining, Inference, and Prediction*. 2nd. Series in Statistics. Berlin: Springer, 2011.

[140] E. Hayashi and J. Hong. "A diary study of password usage in daily life." In: *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '11. Vancouver, BC, Canada: ACM, 2011, pp. 2627–2630.

[141] J. Hennebert. "Speaker recognition, overview." In: *Encyclopedia of Biometrics*. Springer, 2009, pp. 1262–1270.

[142] E. R. Henry. *Classification and uses of finger prints*. HM Stationery Office, 1905.

[143] C. Herley. "So long, and no thanks for the externalities: the rational rejection of security advice by users." In: *Proc. 2009 workshop on New security paradigms workshop*. NSPW '09. Oxford, United Kingdom: ACM, 2009, pp. 133–144.

[144] M. R. Hestbek, C. Nickel, and C. Busch. "Biometric gait recognition for mobile devices using wavelet transform and support vector machines." In: *Proc. Systems, Signals and Image Processing (IWSSIP)*. Apr. 2012, pp. 205–210.

[145] R. Heydon. *Bluetooth Low Energy: The Developer's Handbook*. Prentice Hall, 2012.

[146] K. Hinckley. "Synchronous gestures for multiple persons and computers." In: *Proc. of the 16th Annual ACM Symposium on User Interface Software and Technology*. UIST '03. Vancouver, Canada: ACM, 2003, pp. 149–158.

[147] D. Hintze. "Towards transparent multi-device-authentication." In: *Adjunct Proc. 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proc. 2015 ACM International Symposium on Wearable Computers*. UbiComp/ISWC'15 Adjunct. Osaka, Japan: ACM, 2015, pp. 435–440.

[148] D. Hintze, R. D. Findling, M. Muaaz, E. Koch, and R. Mayrhofer. "Cormorant: towards continuous risk-aware multi-modal cross-device authentication." In: *Proc. 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication (UbiComp 2015)*. Osaka, Japan: ACM, Sept. 2015, pp. 169–172.

[149] D. Hintze, R. D. Findling, M. Muaaz, S. Scholz, and R. Mayrhofer. "Diversity in locked and unlocked mobile device usage." In: *Proc. 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication (UbiComp 2014)*. ACM Press, 2014, pp. 379–384.

[150] D. Hintze, R. D. Findling, S. Scholz, and R. Mayrhofer. "Mobile device usage characteristics: the effect of context and form factor on locked and unlocked usage." In: *Proc. MoMM 2014: 12th International Conference on Advances in Mobile Computing and Multimedia*. Kaohsiung, Taiwan: ACM Press, Dec. 2014, pp. 105–114.

[151] D. Hintze, P. Hintze, R. D. Findling, and R. Mayrhofer. "A large-scale, long-term analysis of mobile device usage characteristics." In: *Proc. ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1.2 (June 2017).

[152] D. Hintze, E. Koch, S. Scholz, and R. Mayrhofer. "Location-based risk assessment for mobile authentication." In: *Adjunct Proc. 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp '16. Heidelberg, Germany: ACM, 2016, pp. 85–88.

[153] D. Hintze, M. Muaaz, R. D. Findling, S. Scholz, E. Koch, and R. Mayrhofer. "Confidence and risk estimation plugins for multi-modal authentication on mobile devices using cormorant." In: *13th International Conference on Advances in Mobile Computing and Multimedia (MoMM 2015)*. Brussels, Belgium: ACM, Dec. 2015, pp. 384–388.

[154] T. Hoang, D. Choi, and T. Nguyen. "Gait authentication on mobile phone using biometric cryptosystem and fuzzy commitment scheme." In: *International Journal of Information Security* (2015), pp. 1–12.

[155] L. E. Holmquist, F. Mattern, B. Schiele, P. Alahuhta, M. Beigl, and H.-W. Gellersen. "Smart-its friends: a technique for users to easily establish connections between smart artefacts." In: *Proc. of the 3rd International Conference on Ubiquitous Computing*. UbiComp '01. Atlanta, Georgia, USA: Springer-Verlag, 2001, pp. 116–122.

[156] M. Hölzl, R. Mayrhofer, and M. Roland. "Requirements for an open ecosystem for embedded tamper resistant hardware on mobile devices." In: *Proc. MoMM 2013*. Vienna, Austria: ACM, 2013, 249:249–249:252.

[157] N. Houmani et al. "BioSecure signature evaluation campaign (BSEC 2009): evaluating online signature algorithms depending on the quality of signatures." In: *Pattern Recognition* 45.3 (2012), pp. 993–1003.

[158] T. Huynh and B. Schiele. "Analyzing features for activity recognition." In: *Proc. of Smart Objects and Ambient Intelligence Soc-EUSAI 2005*. ACM Press, Oct. 2005, pp. 159–163.

[159] R. K. Ibrahim, E. Ambikairajah, B. Celler, N. H. Lovell, and L. Kilmartin. "Gait patterns classification using spectral features." In: *Signals and Systems Conference (ISSC) 2008*. June 2008, pp. 98–102.

[160] Y. Imamverdiyev, A. B. J. Teoh, and J. Kim. "Biometric cryptosystem based on discretized fingerprint texture descriptors." In: *Expert Systems with Applications* 40.5 (2013), pp. 1888–1901.

[161] *Intel Computer Use Research: Usage Tracking Data*. People and Practices Research, Intel Corporation. URL: http://www2.berkeley.intel-research.net/~tlratten/public_usage_data/pud.html.

[162] ISO. *Identification cards – Integrated circuit cards – Part 4: Organization, security and commands for interchange*. 2005.

[163] T. Iso and K. Yamazaki. "Gait analyzer based on a cell phone with a single three-axis accelerometer." In: *Pro. MobileHCI 2006*. Helsinki, Finland: ACM, 2006, pp. 141–144.

[164] A. K. Jain, B. F. Klare, and A. Ross. "Guidelines for best practices in biometrics research." In: *International Conference on Biometrics (ICB)*. Vol. 8. Phuket, Thailand, May 2015.

[165] A. K. Jain and K. Nandakumar. "Biometric authentication: system security and user privacy." In: *IEEE Computer* 45.11 (2012), pp. 87–92.

[166] A. K. Jain, K. Nandakumar, and A. Nagar. "Biometric template security." In: *EURASIP Advances in Signal Processing* 2008 (Jan. 2008), 113:1–113:17.

[167] A. K. Jain, A. A. Ross, and K. Nandakumar. *Introduction to Biometrics*. Springer, 2011.

[168] A. Jain, K. Nandakumar, and A. Ross. "Score normalization in multimodal biometric systems." In: *Pattern recognition* 38.12 (2005), pp. 2270–2285.

[169] W. Jansen. "Authenticating users on handheld devices." In: *Proc. Canadian Information Technology Security Symposium*. 2003, pp. 1–12.

[170] W. Jansen. "Authenticating mobile device users through image selection." In: *WIT Transactions on Information and Communication Technologies* 30 (2004).

[171] W. Jansen, S. Gavrila, and V. Korolev. "Picture password: a visual login technique for mobile devices." In: *NISTIR 7030* (2003).

[172] I. Jermyn, A. J. Mayer, F. Monrose, M. K. Reiter, A. D. Rubin, et al. "The design and analysis of graphical passwords." In: *Usenix Security*. 1999, pp. 1–14.

[173] A. Juels and M. Sudan. "A fuzzy vault scheme." In: *Des. Codes Cryptography* 38.2 (Feb. 2006), pp. 237–257.

[174] A. Juels and M. Wattenberg. "A fuzzy commitment scheme." In: *Proc. 6th ACM Conference on Computer and Communications Security*. CCS '99. Kent Ridge Digital Labs, Singapore: ACM, 1999, pp. 28–36.

[175] T. Kevenaar, G. Schrijen, M. van der Veen, A. Akkermans, and F. Zuo. "Face recognition with renewable and privacy preserving binary templates." In: *Fourth IEEE Workshop on Automatic Identification Advanced Technologies*. Oct. 2005, pp. 21–26.

[176] A. Kholmatov and B. Yanikoglu. "Biometric cryptosystem using online signatures." In: *Computer and Information Sciences – ISCIS 2006*. Ed. by A. Levi, E. Savaş, H. Yenigün, S. Balcısoy, and Y. Saygın. Vol. 4263. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2006, pp. 981–990.

[177] R. Kilian-Kehr. *Securing access to an application service based on a proximity token*. Sept. 2007.

[178] T. Kindberg, C. Bevan, E. O'Neill, J. Mitchell, J. Grimmett, and D. Woodgate. "Authenticating ubiquitous services: a study of wireless hotspot access." In: *Proc. 11th International Conference on Ubiquitous Computing*. UbiComp '09. Orlando, Florida, USA: ACM, 2009, pp. 115–124.

[179] T. Kinnunen and H. Li. "An overview of text-independent speaker recognition: from features to supervectors." In: *Speech Communication* 52.1 (2010), pp. 12–40.

[180] D. Kirovski, M. Sinclair, and D. Wilson. *The Martini Synch*. Tech. rep. MSR-TR-2007-123. Microsoft Research, Sept. 2007.

[181] J. Kittler, Y. Li, and J. Matas. "Face authentication using client specific fisherfaces." In: *The Statistics of Directions, Shapes and Images* (1999), pp. 63–66.

[182] P. Kocher, J. Jaffe, and B. Jun. "Differential power analysis." In: *Proc. CRYPTO99 1999*. 1999, pp. 388–397.

[183] A. Kong, K.-H. Cheung, D. Zhang, M. Kamel, and J. You. "An analysis of biohashing and its variants." In: *Pattern Recognition* 39.7 (July 2006), pp. 1359–1368.

[184] A. Kong, D. Zhang, and M. Kamel. "A survey of palmprint recognition." In: *Pattern Recognition* 42.7 (2009), pp. 1408–1418.

[185] S. G. Kong, J. Heo, B. R. Abidi, J. Paik, and M. A. Abidi. “Recent advances in visual and infrared face recognition—a review.” In: *Computer Vision and Image Understanding* 97.1 (2005), pp. 103–135.

[186] M. Koschuch, M. Hudler, H. Eigner, and Z. Saffer. “Token-based authentication for smartphones.” In: *Data Communication Networking (DCNET), 2013 International Conference on*. July 2013, pp. 1–6.

[187] R. P. Krish, J. Fierrez, J. Galbally, and M. Martinez-Diaz. “Dynamic signature verification on smart phones.” In: *Highlights on Practical Applications of Agents and Multi-Agent Systems: International Workshops of PAAMS 2013*. Ed. by J. M. Corchado, J. Bajo, J. Kozlak, P. Pawlewski, J. M. Molina, V. Julian, R. A. Silveira, R. Unland, and S. Giroux. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 213–222.

[188] M. Kulshrestha, V. Banga, and S. Kumar. “Finger print recognition: survey of minutiae and gabor filtering approach.” In: *International Journal of Computer Applications* 50.4 (2012).

[189] C. Kuo, S. Romanosky, and L. F. Cranor. “Human selection of mnemonic phrase-based passwords.” In: *Proc. Second Symposium on Usable Privacy and Security*. SOUPS ’06. Pittsburgh, Pennsylvania, USA: ACM, 2006, pp. 67–78.

[190] J. R. Kwapisz, G. M. Weiss, and S. A. Moore. “Cell phone-based biometric identification.” In: *Biometrics: Theory Applications and Systems (BTAS) 2010*. Sept. 2010, pp. 1–7.

[191] A. Larcher, K. A. Lee, B. Ma, and H. Li. “Text-dependent speaker verification: classifiers, databases and RSR2015.” In: *Speech Communication* 60 (2014), pp. 56–77.

[192] C. Lee, F. Soong, and K. Paliwal. *Automatic Speech and Speaker Recognition: Advanced Topics*. The Springer International Series in Engineering and Computer Science. Springer US, 2012.

[193] H. Lee, C. Lee, J.-Y. Choi, J. Kim, and J. Kim. “Changeable face representations suitable for human recognition.” In: *Advances in Biometrics*. Ed. by S.-W. Lee and S. Li. Vol. 4642. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, pp. 557–565.

[194] K.-C. Lee, J. Ho, and D. J. Kriegman. “Acquiring linear subspaces for face recognition under variable lighting.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27.5 (May 2005), pp. 684–698.

[195] K. Lee and H. Byun. “A new face authentication system for memory-constrained devices.” In: *IEEE Consumer Electronics* 49.4 (Nov. 2003), pp. 1214–1222.

[196] M. K. Lee, J. O. Park, and J. E. Song. “User authentication based on distance estimation using ultrasonic sensors.” In: *2008 International Conference on Computational Intelligence and Security*. Vol. 2. Dec. 2008, pp. 391–394.

[197] Y. J. Lee, K. R. Park, S. J. Lee, K. Bae, and J. Kim. “A new method for generating an invariant iris private key based on the fuzzy vault system.” In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 38.5 (Oct. 2008), pp. 1302–1313.

[198] J. Lester, B. Hannaford, and G. Borriello. “Are you with me? - using accelerometers to determine if two devices are carried by the same person.” In: *Pervasive*. 2004, pp. 33–50.

[199] F. Li, N. Clarke, M. Papadaki, and P. Dowland. “Active authentication for mobile devices utilising behaviour profiling.” In: *International Journal of Information Security* 13.3 (June 2014), pp. 229–244.

[200] C. Ling and V. Sheng. “Class imbalance problem.” In: *Encyclopedia of Machine Learning*. Ed. by C. Sammut and G. Webb. Springer US, 2010, pp. 171–171.

[201] C. Liu. “Gabor-based kernel PCA with fractional power polynomial models for face recognition.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26.5 (May 2004), pp. 572–581.

[202] M. Long and D. Durham. “Human perceivable authentication: an economical solution for security associations in short-distance wireless networking.” In: *Computer Communications and Networks, 2007. ICCCN 2007. Proceedings of 16th International Conference on*. Aug. 2007, pp. 257–264.

[203] H. Lu, A. J. B. Brush, B. Priyantha, A. K. Karlson, and J. Liu. “Speakersense: energy efficient unobtrusive speaker identification on mobile phones.” In: *Proc. 9th International Conference on Pervasive Computing*. Pervasive’11. San Francisco, USA: Springer-Verlag, 2011, pp. 188–205.

[204] J. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos. “Face recognition using LDA-based algorithms.” In: *IEEE Transactions on Neural Networks* 14.1 (Jan. 2003), pp. 195–200.

[205] L. Ma, B. Wang, S. Narayana, E. Hazeltine, X. Chen, D. A. Robin, and J. Xiong. "Changes in regional activity are accompanied with changes in inter-regional connectivity during four weeks motor learning." In: *Brain Research* 1318.C (2010), pp. 64–76.

[206] U. Mahbub and R. Chellappa. "PATH: person authentication using trace histories." In: *IEEE 7th Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON)*. Oct. 2016, pp. 1–8.

[207] U. Mahbub, S. Sarkar, V. M. Patel, and R. Chellappa. "Active user authentication for smartphones: a challenge data set and benchmark results." In: *IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. Sept. 2016, pp. 1–8.

[208] E. Maiorana and P. Campisi. "Fuzzy commitment for function based signature template protection." In: *Signal Processing Letters, IEEE* 17.3 (Mar. 2010), pp. 249–252.

[209] E. Maiorana, P. Campisi, J. Fierrez, J. Ortega-Garcia, and A. Neri. "Cancelable templates for sequence-based biometrics with application to on-line signature recognition." In: *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* 40.3 (May 2010), pp. 525–538.

[210] E. Maiorana, P. Campisi, and A. Neri. "Template protection for dynamic time warping based biometric signature authentication." In: *16th International Conference on Digital Signal Processing*. July 2009, pp. 1–6.

[211] E. Maiorana, P. Campisi, J. Ortega-Garcia, and A. Neri. "Cancelable biometrics for HMM-based signature recognition." In: *2nd IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS 2008)*. Sept. 2008, pp. 1–6.

[212] F. L. Malallah, S. M. B. S. A. A. Rahman, W. A. B. W. Adnan, and S. B. Yussof. "Article: non-invertible online signature biometric template protection via shuffling and trigonometry transformation." In: *International Journal of Computer Applications* 98.4 (July 2014), pp. 4–17.

[213] N. Malkin, M. Harbach, A. De Luca, and S. Egelman. "The anatomy of smartphone unlocking: why and how Android users around the world lock their phones." In: *GetMobile: Mobile Comp. and Comm.* 20.3 (Jan. 2017), pp. 42–46.

[214] D. Maltoni, D. Maio, A. Jain, and S. Prabhakar. *Handbook of Fingerprint Recognition*. Springer Professional Computing. Springer London, 2009.

[215] A. Mannini and A. M. Sabatini. "Machine learning methods for classifying human physical activity from on-body accelerometers." In: *Sensors* 10.2 (2010), pp. 1154–1175.

[216] J. Mäntyjärvi, M. Lindholm, E. Vildjiounaite, S.-M. Mäkelä, and H. Ailisto. "Identifying users of portable devices from gait pattern with accelerometers." In: *Proc. IEEE Acoustics, Speech, and Signal Processing (ICASSP) 2005*. Vol. 2. 2005, pp. ii–973.

[217] E. Marasco and A. Ross. "A survey on antispoofing schemes for fingerprint recognition systems." In: *ACM Computuing Surveys (CSUR)* 47.2 (Nov. 2014), 28:1–28:36.

[218] S. Marcel, C. McCool, P. Matejka, T. Ahonen, and J. Cernocky. *Mobile biometry (MOBIO) face and speaker verification evaluation*. Tech. rep. Idiap, 2010.

[219] S. Marcel et al. "On the results of the first mobile biometry (MOBIO) face and speaker verification evaluation." In: *Proc. 20th International Conference on Recognizing Patterns in Signals, Speech, Images, and Videos*. ICPR'10. Istanbul, Turkey: Springer-Verlag, 2010, pp. 210–225.

[220] R. Marin-Perianu, M. Marin-Perianu, P. Havinga, and H. Scholten. "Movement-based group awareness with wireless sensor networks." In: *Proc. of the 5th International Conference on Pervasive Computing, Pervasive'07*. Toronto, Canada: Springer-Verlag, 2007, pp. 298–315.

[221] M. D. Marsico, M. Nappi, D. Riccio, and H. Wechsler. "Mobile iris challenge evaluation (MICHE)-I, biometric iris dataset and protocols." In: *Pattern Recognition Letters* 57 (2015), pp. 17–23.

[222] M. Martinez-Diaz, J. Fierrez-Aguilar, F. Alonso-Fernandez, J. Ortega-Garcia, and J. Siguenza. "Hill-climbing and brute-force attacks on biometric systems: a case study in match-on-card fingerprint verification." In: *Proc. IEEE Security Technology*. Oct. 2006, pp. 151–159.

[223] L. Masek. *Recognition of Human Iris Patterns for Biometric Identification*. Tech. rep. University of Western Australia, 2003.

[224] R. Mayrhofer and H. Gellersen. "Shake well before use: intuitive and secure pairing of mobile devices." In: *IEEE Transactions on Mobile Computing* 8.6 (2009), pp. 792–806.

[225] R. Mayrhofer. "The candidate key protocol for generating secret shared keys from similar sensor data streams." In: *Proc. of the 4th European Conference on Security and Privacy in Ad-hoc and Sensor Networks*. ESAS'07. Cambridge, UK: Springer-Verlag, 2007, pp. 1–15.

[226] R. Mayrhofer. "An architecture for secure mobile devices." In: *Security and Communication Networks* 8 (July 2014), pp. 1958–1970.

[227] R. Mayrhofer, J. Fuss, and I. Ion. "UACAP: a unified auxiliary channel authentication protocol." In: *IEEE Transactions on Mobile Computing* 12.4 (Apr. 2013), pp. 710–721.

[228] R. Mayrhofer and H. Gellersen. "On the security of ultrasound as out-of-band channel." In: *Parallel and Distributed Processing Symposium (IPDPS 2007)*. IEEE. 2007, pp. 1–6.

[229] R. Mayrhofer, H. Hlavacs, and R. D. Findling. "Optimal derotation of shared acceleration time series by determining relative spatial alignment." In: *Proc. iiWAS 2014: 16th International Conference on Information Integration and Web-based Applications & Services*. Hanoi, Vietnam: ACM Press, Dec. 2014, pp. 71–78.

[230] R. Mayrhofer, H. Hlavacs, and R. D. Findling. "Optimal derotation of shared acceleration time series by determining relative spatial alignment." In: *International Journal of Pervasive Computing and Communications (IJPCC)* 11.4 (Oct. 2015), pp. 454–466.

[231] W. Meng, D. S. Wong, S. Furnell, and J. Zhou. "Surveying the development of biometric user authentication on mobile phones." In: *IEEE Communications Surveys Tutorials* 17.3 (2015), pp. 1268–1293.

[232] J. Merkle, M. Niesing, M. Schwaiger, H. Ihmor, and U. Korte. "Provable security for the fuzzy fingerprint vault." In: *Internet Monitoring and Protection (ICIMP), 2010 Fifth International Conference on*. May 2010, pp. 65–73.

[233] L. Middleton, A. A. Buss, A. Bazin, and M. S. Nixon. "A floor sensor system for gait recognition." In: *Proc. IEEE Automatic Identification Advanced Technologies (AutoID) 2005*. Oct. 2005, pp. 171–176.

[234] J. Ming, T. J. Hazen, J. R. Glass, and D. A. Reynolds. "Robust speaker recognition in noisy conditions." In: *IEEE Transactions on Audio, Speech, and Language Processing* 15.5 (July 2007), pp. 1711–1723.

[235] F. Monrose, M. K. Reiter, Q. Li, and S. Wetzel. "Cryptographic key generation from voice." In: *Proc. 2001 IEEE Symposium on Security and Privacy*. SP '01. Washington, DC, USA: IEEE Computer Society, 2001, pp. 202–211.

[236] H. M. Moon, C. Won, and S. B. Pan. "The multi-modal human identification based on smartcard in video surveillance system." In: *Proc. IEEE/ACM GreenCom and CPSCom 2010*. Dec. 2010, pp. 691–698.

[237] B. Morrow. "BYOD security challenges: control and protect your most sensitive data." In: *Network Security* 2012.12 (2012), pp. 5–8.

[238] A. Mostayed, S. Kim, M. M. G. Mazumder, and S. J. Park. "Foot step based person identification using histogram similarity and wavelet decomposition." In: *Proc. Information Security and Assurance (ISA) 2008*. Apr. 2008, pp. 307–311.

[239] M. Muaaz and R. Mayrhofer. "Smartphone-based gait recognition: from authentication to imitation." In: *IEEE Transactions on Mobile Computing* (2017).

[240] M. Muaaz and R. Mayrhofer. "An analysis of different approaches to gait recognition using cell phone based accelerometers." In: *Proc. MoMM 2013*. Vienna, Austria: ACM, 2013, 293:293–293:300.

[241] M. Muaaz and R. Mayrhofer. "Orientation independent cell phone based gait authentication." In: *Proc. MoMM 2014*. Kaohsiung, Taiwan: ACM, 2014, pp. 161–164.

[242] M. Muaaz and R. Mayrhofer. "Cross pocket gait authentication using mobile phone based accelerometer sensor." In: *Proc. Computer Aided Systems Theory (EUROCAST) 2015*. Las Palmas de Gran Canaria, Spain: Springer, Feb. 2015, pp. 731–738.

[243] M. Muaaz and R. Mayrhofer. "Accelerometer based gait recognition using adapted gaussian mixture models." In: *Proc. 14th International Conference on Advances in Mobile Computing and Multimedia (MoMM 2016)*. ACM. Singapore: ACM, Nov. 2016, pp. 288–291.

[244] P. M. Murray, B. A. Drought, and R. C. Kory. "Walking Patterns of Normal Men." In: *The Journal of Bone & Joint Surgery* 46.2 (Mar. 1964), pp. 335–360.

[245] I. Muslukhov, Y. Boshmaf, C. Kuo, J. Lester, and K. Beznosov. "Understanding users' requirements for data protection in smartphones." In: *Proc. ICDEW 2012*. 2012, pp. 228–235.

[246] A. Nagar, K. Nandakumar, and A. Jain. "Securing fingerprint template: fuzzy vault with minutiae descriptors." In: *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*. Dec. 2008, pp. 1–4.

[247] A. Nagar, K. Nandakumar, and A. Jain. "Multibiometric cryptosystems based on feature-level fusion." In: *IEEE Transactions on Information Forensics and Security* 7.1 (Feb. 2012), pp. 255–268.

[248] A. Nagar. "Biometric template security." PhD thesis. Michigan State University, 2012.

[249] A. Nagar, K. Nandakumar, and A. K. Jain. "A hybrid biometric cryptosystem for securing fingerprint minutiae templates." In: *Pattern Recognition Letters* 31.8 (June 2010), pp. 733–741.

[250] K. Nandakumar. "A fingerprint cryptosystem based on minutiae phase spectrum." In: *IEEE International Workshop on Information Forensics and Security (WIFS)*. Dec. 2010, pp. 1–6.

[251] M. A. Nematollahi and S. Al-Haddad. "Distant speaker recognition: an overview." In: *International Journal of Humanoid Robotics* 13.02 (2016), p. 1550032.

[252] D. C. L. Ngo, A. B. J. Teoh, and J. Hu. *Biometric Security*. Cambridge Scholars Publishing, 2015.

[253] A. J. Nicholson, M. D. Corner, and B. D. Noble. "Mobile device security using transient authentication." In: *IEEE Transactions on Mobile Computing* 5.11 (Nov. 2006), pp. 1489–1502.

[254] C. Nickel. "Accelerometer-based Biometric Gait Recognition for Authentication on Smartphones." PhD thesis. Technische Universität Darmstadt, 2012.

[255] V. Niennattrakul and C. A. Ratanamahatana. "Learning DTW global constraint for time series classification." In: *CoRR* abs/0903.0041 (2009).

[256] L. O'Gorman. "Comparing passwords, tokens, and biometrics for user authentication." In: *Proc. IEEE* 91.12 (Dec. 2003), pp. 2021–2040.

[257] S. Ojala, J. Keinanen, and J. Skytta. "Wearable authentication device for transparent login in nomadic applications environment." In: *2008 2nd International Conference on Signals, Circuits and Systems*. Nov. 2008, pp. 1–6.

[258] P. C. van Oorschot, A. Salehi-Abari, and J. Thorpe. "Purely automated attacks on passpoints-style graphical passwords." In: *IEEE Transactions on Information Forensics and Security* 5.3 (Sept. 2010), pp. 393–405.

[259] O. Ouda, N. Tsumura, and T. Nakaguchi. "Tokenless cancelable biometrics scheme for protecting iris codes." In: *Pattern Recognition (ICPR), 2010 20th International Conference on*. Aug. 2010, pp. 882–885.

[260] S. B. Pan, D. Moon, Y. Gil, D. Ahn, and Y. Chung. "An ultra-low memory fingerprint matching algorithm and its implementation on a 32-bit smart card." In: *IEEE Consumer Electronics* 49.2 (May 2003), pp. 453–459.

[261] K. R. Park, H.-A. Park, B. J. Kang, E. C. Lee, and D. S. Jeong. "A study on iris localization and recognition on mobile phones." In: *EURASIP Journal on Advances in Signal Processing* 2008 (Jan. 2008).

[262] V. M. Patel, R. Chellappa, D. Chandra, and B. Barbello. "Continuous user authentication on mobile devices: recent progress and remaining challenges." In: *IEEE Signal Processing Magazine* 33.4 (July 2016), pp. 49–61.

[263] V. Patel, N. Ratha, and R. Chellappa. "Cancelable biometrics: a review." In: *Signal Processing Magazine, IEEE* 32.5 (Sept. 2015), pp. 54–65.

[264] R. Plamondon and S. N. Srihari. "Online and off-line handwriting recognition: a comprehensive survey." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.1 (Jan. 2000), pp. 63–84.

[265] H. Proença and L. A. Alexandre. "UBIRIS: a noisy iris image database." In: *13th International Conference on Image Analysis and Processing (ICIAP 2005)*. Ed. by F. Roli and S. Vitulano. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 970–977.

[266] T. van der Putte and J. Keuning. "Biometrical fingerprint recognition: don't get your fingers burned." In: *Smart Card Research and Advanced Applications: IFIP TC8 / WG8.8 Fourth Working Conference on Smart Card Research and Advanced Applications September 20–22, 2000, Bristol, United Kingdom*. Ed. by J. Domingo-Ferrer, D. Chan, and A. Watson. Boston, MA: Springer US, 2000, pp. 289–303.

[267] S. Rahati, R. Moravejian, and F. M. Kazemi. "Gait recognition using wavelet transform." In: *Proc. Information Technology: New Generations (ITNG) 2008*. Apr. 2008, pp. 932–936.

[268] K. B. Raja, R. Raghavendra, and C. Busch. "Iris imaging in visible spectrum using white led." In: *2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. Sept. 2015, pp. 1–8.

[269] W. Rankl and W. Effing. *Smart Card Handbook*. Wiley, 2004.

[270] A. Rao, B. Jha, and G. Kini. "Effect of grammar on security of long passwords." In: *Proc. third ACM conference on Data and application security and privacy*. CODASPY '13. San Antonio, Texas, USA: ACM, 2013, pp. 317–324.

[271] K. S. Rao and S. Sarkar. "Robust speaker verification: a review." In: *Robust Speaker Recognition in Noisy Environments*. Springer, 2014, pp. 13–27.

[272] N. K. Ratha, J. H. Connell, and R. M. Bolle. "Enhancing security and privacy in biometrics-based authentication systems." In: *IBM Systems Journal* 40.3 (Mar. 2001), pp. 614–634.

[273] N. Ratha and R. Bolle. *Automatic Fingerprint Recognition Systems*. Springer New York, 2007.

[274] N. K. Ratha. "Privacy protection in high security biometrics applications." In: *Ethics and Policy of Biometrics*. Ed. by A. Kumar and D. Zhang. Vol. 6005. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2010, pp. 62–69.

[275] N. K. Ratha, J. H. Connell, and R. M. Bolle. "An analysis of minutiae matching strength." In: *Proc. Third International Conference on Audio- and Video-Based Biometric Person Authentication*. AVBPA '01. London, UK, UK: Springer-Verlag, 2001, pp. 223–228.

[276] N. Ratha and V. Govindaraju. *Advances in Biometrics: Sensors, Algorithms and Systems*. Springer London, 2007.

[277] C. Rathgeb, F. Breitinger, and C. Busch. "Alignment-free cancelable iris biometric templates based on adaptive bloom filters." In: *Biometrics (ICB), 2013 International Conference on*. June 2013, pp. 1–8.

[278] C. Rathgeb. "Iris Based Biometric Cryptosystems." MA thesis. Jakob Haringer Strasse 2, 5020 Salzburg, Austria: Department of Computer Sciences, University of Salzburg, Nov. 2008.

[279] C. Rathgeb, F. Breitinger, C. Busch, and H. Baier. "On application of bloom filters to iris biometrics." In: *IET Biometrics* (Jan. 2014).

[280] C. Rathgeb and C. Busch. "Multi-biometric template protection: issues and challenges." In: *New Trends and Developments in Biometrics*. Ed. by J. Yang and S. J. Xie. Communications and Security. InTech, Nov. 2012.

[281] C. Rathgeb and A. Uhl. "A survey on biometric cryptosystems and cancelable biometrics." In: *EURASIP Journal on Information Security* 2011.1 (2011), pp. 1–25.

[282] H. Ravi and S. K. Sivanath. "A novel method for touch-less finger print authentication." In: *2013 IEEE International Conference on Technologies for Homeland Security (HST)*. IEEE. 2013, pp. 147–153.

[283] J. Rekimoto. "SyncTap: synchronous user operation for spontaneous network connection." In: *Personal and Ubiquitous Computing* 8.2 (May 2004), pp. 126–134.

[284] D. A. Reynolds. "An overview of automatic speaker recognition technology." In: *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 4. May 2002, pp. 4072–4075.

[285] P. Riedl, R. Mayrhofer, A. Möller, M. Kranz, F. Lettner, C. Holzmann, and M. Koelle. "Only play in your comfort zone: interaction methods for improving security awareness on mobile devices." In: *Personal and Ubiquitous Computing* 19.5 (Aug. 2015), pp. 941–954.

[286] O. Riva, C. Qin, K. Strauss, and D. Lymberopoulos. "Progressive authentication: deciding when to authenticate on mobile phones." In: *Presented as part of the 21st USENIX Security Symposium (USENIX Security 12)*. Bellevue, WA: USENIX, 2012, pp. 301–316.

[287] R. L. Rivest and A. Shamir. "How to expose an eavesdropper." In: *Communications of the ACM* 27.4 (Apr. 1984), pp. 393–394.

[288] P. Roberts, L. Benofsky, W. Holt, L. Johnson, M. Bryant, and N. Nussbaum. *Systems and methods for demonstrating authenticity of a virtual machine using a security image*. July 2009.

[289] P. Roberts, L. Benofsky, W. Holt, L. Johnson, B. Willman, and M. Bryant. *Systems and methods for determining if applications executing on a computer system are trusted*. May 2010.

[290] M. Roland. *Security issues in mobile NFC devices*. Springer, 2013.

[291] L. Rong, Z. Jianzhong, L. Ming, and H. Xiangfeng. "A wearable acceleration sensor system for gait recognition." In: *Proc. Industrial Electronics and Applications*. May 2007, pp. 2654–2659.

[292] A. Ross, J. Shah, and A. Jain. "From template to image: reconstructing fingerprints from minutiae points." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.4 (Apr. 2007), pp. 544–560.

[293] A. A. Ross, K. Nandakumar, and A. K. Jain. *Handbook of Multibiometrics (International Series on Biometrics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[294] N. Roy, M. Gowda, and R. R. Choudhury. "Ripple: communicating through physical vibration." In: *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15)*. Oakland, CA: USENIX Association, May 2015, pp. 265–278.

[295] A. P. Sabzevar and A. Stavrou. "Universal multi-factor authentication using graphical passwords." In: *2008 IEEE International Conference on Signal Image Technology and Internet Based Systems*. Nov. 2008, pp. 625–632.

[296] H. Saevanee, N. Clarke, S. Furnell, and V. Biscione. "Text-based active authentication for mobile devices." In: *29th IFIP TC 11 International Conference on ICT Systems Security and Privacy Protection (SEC 2014)*. Ed. by N. Cuppens-Boulahia, F. Cuppens, S. Jajodia, A. Abou El Kalam, and T. Sans. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 99–112.

[297] H. Sakoe and S. Chiba. "Dynamic programming algorithm optimization for spoken word recognition." In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 26.1 (Feb. 1978), pp. 43–49.

[298] P. Samangouei, V. M. Patel, and R. Chellappa. "Attribute-based continuous user authentication on mobile devices." In: *2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. Sept. 2015, pp. 1–8.

[299] S. Sarkar, P. J. Phillips, Z. Liu, I. R. Vega, P. Grother, and K. W. Bowyer. "The humanid gait challenge problem: data sets, performance, and analysis." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27.2 (Feb. 2005), pp. 162–177.

[300] M. Sasse, S. Brostoff, and D. Weirich. "Transforming the 'weakest link' — a human/-computer interaction approach to usable and effective security." In: *BT Technology Journal* 19.3 (2001), pp. 122–131.

[301] A. Savitzky and M. J. E. Golay. "Smoothing and differentiation of data by simplified least squares procedures." In: *Analytical Chemistry* 36.8 (1964), pp. 1627–1639.

[302] N. Saxena, M. B. Uddin, J. Voris, and N. Asokan. "Vibrate-to-unlock: mobile phone assisted user authentication to multiple personal RFID tags." In: *2011 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. Mar. 2011, pp. 181–188.

[303] E. S. Sazonov, T. Bumpus, S. Zeigler, and S. Marocco. "Classification of plantar pressure and heel acceleration patterns using neural networks." In: *Proc. Neural Networks 2005*. Vol. 5. July 2005, pp. 3007–3010.

[304] F. Schaub, R. Deyhle, and M. Weber. "Password entry usability and shoulder surfing susceptibility on different smartphone platforms." In: *Proc. of the 11th International Conference on Mobile and Ubiquitous Multimedia*. MUM '12. Ulm, Germany: ACM, 2012, 13:1–13:10.

[305] F. Schaub, M. Walch, B. Könings, and M. Weber. "Exploring the design space of graphical passwords on smartphones." In: *Proc. Ninth Symposium on Usable Privacy and Security*. SOUPS '13. Newcastle, United Kingdom: ACM, 2013, 11:1–11:14.

[306] R. Schlöglhofer and J. Sametinger. "Secure and usable authentication on mobile devices." In: *Proc. 10th International Conference on Advances in Mobile Computing and Multimedia*. MoMM '12. Bali, Indonesia: ACM, 2012, pp. 257–262.

[307] R. A. Schmidt and T. D. Lee. *Motor control and learning: A behavioral emphasis, 5th edition*. Vol. 4. Human Kinetics, Mar. 2011.

[308] T. Schmidt, V. Riffo, and D. Mery. "Dynamic signature recognition based on fisher discriminant." In: *16th Iberoamerican Congress on Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications (CIARP 2011)*. Ed. by C. San Martin and S.-W. Kim. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 433–442.

[309] S. Schneegass, F. Steimle, A. Bulling, F. Alt, and A. Schmidt. "SmudgeSafe: geometric image transformations for smudge-resistant user authentication." In: *Proc. 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp '14. Seattle, Washington: ACM, 2014, pp. 775–786.

[310] D. Schürmann, A. Brüsch, S. Sigg, and L. Wolf. "BANDANA - body area network device-to-device authentication using natural gait." In: *2017 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. Mar. 2017, pp. 190–196.

[311] D. R. Shanks and M. F. St. John. "Characteristics of dissociable human learning systems." In: *Behavioral and Brain Sciences* 17.3 (1994), pp. 367–395.

[312] C. E. Shannon. "A mathematical theory of communication." In: *The Bell System Technical Journal* 27.3 (July 1948), pp. 379–423.

[313] R. Singh, M. Vatsa, A. Ross, and A. Noore. "A mosaicing scheme for pose-invariant face recognition." In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 37.5 (Oct. 2007), pp. 1212–1225.

[314] L. Sirovich and M. Kirby. "Low-Dimensional Procedure for the Characterization of Human Faces." In: *Journal of the Optical Society of America A* 4.3 (1987), pp. 519–524.

[315] R. E. Smith. *Authentication: From Passwords to Public Keys*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2002.

[316] M. Smith-Creasey and M. Rajarajan. "A continuous user authentication scheme for mobile devices." In: *2016 14th Annual Conference on Privacy, Security and Trust (PST)*. Dec. 2016, pp. 104–113.

[317] Y. Song, G. Cho, S. Oh, H. Kim, and J. H. Huh. "On the effectiveness of pattern lock strength meters: measuring the strength of real world pattern locks." In: *Proc. 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. Seoul, Republic of Korea: ACM, 2015, pp. 2343–2352.

[318] C. Soriente, G. Tsudik, and E. Uzun. "BEDA: button-enabled device pairing." In: *IACR Cryptology ePrint Archive* 2007 (2007), p. 246.

[319] C. Soriente, G. Tsudik, and E. Uzun. "HAPADEP: human-assisted pure audio device pairing." In: *Information Security*. Ed. by T.-C. Wu, C.-L. Lei, V. Rijmen, and D.-T. Lee. Vol. 5222. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2008, pp. 385–400.

[320] C. Sousedik and C. Busch. "Presentation attack detection methods for fingerprint recognition systems: a survey." In: *IET Biometrics* 3 (4 Dec. 2014), 219–233(14).

[321] S. Sprager and D. Zazula. "A cumulant-based method for gait identification using accelerometer data with principal component analysis and support vector machine." In: *WSEAS Transactions on Signal Processing* 5.11 (Nov. 2009), pp. 369–378.

[322] F. Stajano. "Pico: no more passwords!" In: *Security Protocols XIX: 19th International Workshop, Cambridge, UK, March 28-30, 2011, Revised Selected Papers*. Ed. by B. Christianson, B. Crispo, J. Malcolm, and F. Stajano. Vol. 7114. Security Protocols 2011: Security Protocols XIX, Lecture Notes in Computer Science (LNCS). Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 49–81.

[323] K. Stefan, L. G. Cohen, J. Duque, R. Mazzocchio, P. Celnik, L. Sawaki, L. Ungerleider, and J. Classen. "Formation of a motor memory by action observation." In: *Journal of Neuroscience* 25.41 (2005), pp. 9339–9346.

[324] A. Studer, T. Passaro, and L. Bauer. "Don't bump, shake on it: the exploitation of a popular accelerometer-based smart phone exchange and its secure replacement." In: *Proc. of the 27th Annual Computer Security Applications Conference*. ACSAC '11. Orlando, Florida: ACM, 2011, pp. 333–342.

[325] D. Z. Sun, J. P. Huai, J. Z. Sun, J. W. Zhang, and Z. Y. Feng. "A new design of wearable token system for mobile device security." In: *IEEE Transactions on Consumer Electronics* 54.4 (Nov. 2008), pp. 1784–1789.

[326] X. Suo, Y. Zhu, and G. S. Owen. "Graphical passwords: a survey." In: *Proc. 21st Annual Computer Security Applications Conference*. ACSAC '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 463–472.

[327] Y. Sutcu, Q. Li, and N. Memon. "Protecting biometric templates with sketch: theory and practice." In: *IEEE Transactions on Information Forensics and Security* 2.3 (Sept. 2007), pp. 503–512.

[328] Y. Sutcu, Q. Li, and N. Memon. "Secure biometric templates from fingerprint-face features." In: *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*. June 2007, pp. 1–6.

[329] M. Swan. "Sensor mania! the internet of things, wearable computing, objective metrics, and the quantified self 2.0." In: *Journal of Sensor and Actuator Networks (JSAN)* 1.3 (Nov. 2012), pp. 217–253.

[330] B. Tams, J. Merkle, C. Rathgeb, J. Wagner, U. Korte, and C. Busch. "Improved fuzzy vault scheme for alignment-free fingerprint features." In: *2015 International Conference of the Biometrics Special Interest Group (BIOSIG)*. Sept. 2015, pp. 1–12.

[331] M. Tamviruzzaman, S. I. Ahamed, C. S. Hasan, and C. O'brien. "ePet: when cellular phone learns to recognize its owner." In: *Proc. Assurable and Usable Security Configuration (SafeConfig) 2009*. Chicago, Illinois, USA: ACM, 2009, pp. 13–18.

[332] P. Tanvi, G. Sonal, and S. M. Kumar. "Token based authentication using mobile phone." In: *2011 International Conference on Communication Systems and Network Technologies*. June 2011, pp. 85–88.

[333] H. Tao. "Pass-Go, a new graphical password scheme." PhD thesis. University of Ottawa (Canada), 2006.

[334] F. Tari, A. A. Ozok, and S. H. Holden. "A comparison of perceived and real shoulder-surfing risks between alphanumeric and graphical passwords." In: *Proc. of the second symposium on Usable privacy and security*. SOUPS '06. Pittsburgh, Pennsylvania: ACM, 2006, pp. 56–66.

[335] A. Teoh, A. Goh, and D. Ngo. "Random multispace quantization as an analytic mechanism for BioHashing of biometric and random identity inputs." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.12 (Dec. 2006), pp. 1892–1901.

[336] D. van Thanh, I. Jorstad, T. Jonvik, and D. van Thuan. "Strong authentication with mobile phone as security token." In: *2009 IEEE 6th International Conference on Mobile Adhoc and Sensor Systems*. Oct. 2009, pp. 777–782.

[337] G. Thomson. "BYOD: enabling the chaos." In: *Network Security* 2012.2 (2012), pp. 5–8.

[338] J. Thorpe and P. C. van Oorschot. "Towards secure design choices for implementing graphical passwords." In: *20th Annual Computer Security Applications Conference*. Dec. 2004, pp. 50–60.

[339] J. Thorpe and P. C. van Oorschot. "Graphical dictionaries and the memorable space of graphical passwords." In: *USENIX Security Symposium*. 2004, pp. 135–150.

[340] J. Thorpe and P. C. van Oorschot. "Human-seeded attacks and exploiting hot-spots in graphical passwords." In: *Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium*. SS'07. Boston, MA: USENIX Association, 2007, 8:1–8:16.

[341] H. C. A. van Tilborg and S. Jajodia, eds. *Encyclopedia of Cryptography and Security, 2nd Ed.* Springer, 2011.

[342] M. Tistarelli and E. Grosso. "Active vision-based face authentication." In: *Image and Vision Computing* 18.4 (2000), pp. 299–314.

[343] R. Tolosana, R. Vera-Rodriguez, J. Ortega-Garcia, and J. Fierrez. "Optimal feature selection and inter-operability compensation for on-line biometric signature authentication." In: *2015 International Conference on Biometrics (ICB)*. May 2015, pp. 163–168.

[344] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia. "Benchmarking desktop and mobile handwriting across COTS devices: the e-BioSign biometric database." In: *PLoS ONE* 12.5 (May 2017), pp. 1–17.

[345] V. Tong, H. Sibert, J. Lecœur, and M. Girault. "Biometric fuzzy extractors made practical: a proposal based on fingercodes." In: *Advances in Biometrics*. Ed. by S.-W. Lee and S. Li. Vol. 4642. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, pp. 604–613.

[346] P. Tresadern, T. Cootes, N. Poh, P. Matejka, A. Hadid, C. Lévy, C. McCool, and S. Marcel. "Mobile biometrics: combined face and voice verification for a mobile platform." In: *IEEE Pervasive Computing* 12.1 (2013), pp. 79–87.

[347] U. Uludag and A. K. Jain. "Attacks on biometric systems: a case study in fingerprints." In: *Security, Steganography, and Watermarking of Multimedia Contents VI*. Vol. 5306. Society of Photo-Optical Instrumentation Engineers (SPIE). June 2004, pp. 622–633.

[348] U. Uludag, S. Pankanti, S. Prabhakar, and A. Jain. "Biometric cryptosystems: issues and challenges." In: *Proc. IEEE* 92.6 (June 2004), pp. 948–960.

[349] U. Uludag. "Secure Biometric Systems." PhD thesis. Michigan State University, 2006.

[350] D. Van Bruggen, S. Liu, M. Kajzer, A. Striegel, C. R. Crowell, and J. D'Arcy. "Modifying smartphone user locking behavior." In: *Proc. SOUPS 2013*. Newcastle, United Kingdom: ACM, 2013, 10:1–10:14.

[351] C. Varenhorst, M. Kleek, and L. Rudolph. "Passdoodles: a lightweight authentication method." In: *Research Science Institute* (2004).

[352] C. L. Vaughan, B. L. Davis, and J. C. O'Connor. *Dynamics of Human Gait*. Ed. by C. L. Vaughan. Second. Howard Place, Western Cape 7450, South Africa: Kiboho Publishers, 1999.

[353] S. Venugopalan and M. Savvides. "How to generate spoofed irises from an iris code template." In: *IEEE Transactions on Information Forensics and Security* 6.2 (June 2011), pp. 385–395.

[354] D. Vermoen, M. Witteman, and G. N. Gaydadjiev. "Reverse engineering Java Card applets using power analysis." In: *Proc. IFIP 2007*. Springer, 2007, pp. 138–149.

[355] B. Vibert, C. Rosenberger, and A. Ninassi. "Security and performance evaluation platform of biometric match on card." In: *2013 World Congress on Computer and Information Technology (WCCIT)*. June 2013, pp. 1–6.

[356] P. Viola and M. Jones. "Robust real-time face detection." In: *International Journal of Computer Vision* 57 (2004), pp. 137–154.

[357] J. Wang, K. Plataniotis, and A. Venetsanopoulos. "Selecting discriminant eigenfaces for face recognition." In: *Pattern Recognition Letters* 26.10 (2005), pp. 1470–1482.

[358] X. Wang, Y. Li, and F. Qiao. "Gait authentication based on multi-criterion model of acceleration features." In: *Proc. Modelling, Identification and Control (ICMIC) 2010*. July 2010, pp. 664–669.

[359] Y. Wang and K. Plataniotis. "Face based biometric authentication with changeable and privacy preservable templates." In: *Biometrics Symposium, 2007*. Sept. 2007, pp. 1–6.

[360] M. Weiser. "The computer for the 21st century." In: *Scientific american* 265.3 (1991), pp. 94–104.

[361] K. Weiss. *Method and apparatus for positively identifying an individual*. Jan. 1988.

[362] R. Weiss and A. De Luca. "PassShapes: utilizing stroke based authentication to increase password memorability." In: *Proc. 5th Nordic Conference on Human-computer Interaction: Building Bridges*. NordiCHI '08. Lund, Sweden: ACM, 2008, pp. 383–392.

[363] P. D. Welch. "The use of fast Fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms." In: *IEEE Transactions on Audio and Electroacoustics* 15.2 (1967), pp. 70–73.

[364] M. Whittle. *Gait analysis: an introduction*. 3rd ed. Elsevier, 2002.

[365] S. Wiedenbeck, J. Waters, J.-C. Birget, A. Brodskiy, and N. Memon. "Authentication using graphical passwords: effects of tolerance and image choice." In: *Proc. 2005 Symposium on Usable Privacy and Security*. SOUPS '05. Pittsburgh, Pennsylvania, USA: ACM, 2005, pp. 1–12.

[366] S. Wiedenbeck, J. Waters, J.-C. Birget, A. Brodskiy, and N. Memon. "PassPoints: design and longitudinal evaluation of a graphical password system." In: *International Journal of Human-Computer Studies* 63.1-2 (July 2005), pp. 102–127.

[367] S. Wiedenbeck, J. Waters, L. Sobrado, and J.-C. Birget. "Design and evaluation of a shoulder-surfing resistant graphical password scheme." In: *Proc. Working Conference on Advanced Visual Interfaces*. AVI '06. Venezia, Italy: ACM, 2006, pp. 177–184.

[368] D. Winter. *Biomechanics and Motor Control of Human Movement*. Wiley, 2004.

[369] W. Wodo and S. Zientek. "Biometric linkage between identity document card and its holder based on real-time facial recognition." In: *Science and Information Conference (SAI), 2015*. July 2015, pp. 1380–1383.

[370] D. M. Wolpert, Z. Ghahramani, and J. Flanagan. "Perspectives and problems in motor learning." In: *Trends in Cognitive Sciences* 5.11 (2001), pp. 487–494.

[371] K. Xi and J. Hu. "Biometric mobile template protection: a composite feature based fingerprint fuzzy vault." In: *IEEE International Conference on Communications (ICC 2009)*. June 2009, pp. 1–5.

[372] K. Xi, J. Hu, and F. Han. "An alignment free fingerprint fuzzy extractor using near-equivalent dual layer structure check (nedlsc) algorithm." In: *2011 6th IEEE Conference on Industrial Electronics and Applications (ICIEA)*. June 2011, pp. 1040–1045.

[373] N. Yager and A. Amin. "Fingerprint classification: a review." In: *Pattern Analysis and Applications* 7.1 (2004), pp. 77–93.

[374] S. Yang and I. Verbauwhede. "Automatic secure fingerprint verification system based on fuzzy vault scheme." In: *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*. Vol. 5. Mar. 2005, pp. 609–612.

[375] T. Yonezawa, J. Nakazawa, and H. Tokuda. "Vinteraction: vibration-based information transfer for smart devices." In: *Mobile Computing and Ubiquitous Networking (ICMU), 2015 Eighth International Conference on*. Jan. 2015, pp. 155–160.

[376] S. Yoon. *Fingerprint recognition: models and applications*. Michigan State University, 2014.

[377] E. von Zezschwitz, P. Dunphy, and A. De Luca. "Patterns in the wild: a field study of the usability of pattern and pin-based authentication on mobile devices." In: *Proc. of the 15th international conference on Human-computer interaction with mobile devices and services*. MobileHCI '13. Munich, Germany: ACM, 2013, pp. 261–270.

[378] E. von Zezschwitz, A. Koslow, A. De Luca, and H. Hussmann. "Making graphic-based authentication secure against smudge attacks." In: *Proc. of the 2013 international conference on Intelligent user interfaces*. Santa Monica, California, USA: ACM, 2013, pp. 277–286.

[379] E. Zezschwitz, A. Luca, and H. Hussmann. "Survival of the shortest: a retrospective analysis of influencing factors on password composition." In: *Human-Computer Interaction (INTERACT 2013)*. Ed. by P. Kotzé, G. Marsden, G. Lindgaard, J. Wesson, and M. Winckler. Vol. 8119. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2013, pp. 460–467.

[380] X. Zhang and Y. Gao. "Face recognition across pose: a review." In: *Pattern Recognition* 42.11 (Nov. 2009), pp. 2876–2896.

[381] L. Zhang-Kennedy, S. Chiasson, and P. van Oorschot. "Revisiting password rules: facilitating human management of passwords." In: *2016 APWG Symposium on Electronic Crime Research (eCrime)*. June 2016, pp. 1–10.

[382] X. Zou, J. Kittler, and K. Messer. "Illumination invariant face recognition: a survey." In: *2007 First IEEE International Conference on Biometrics: Theory, Applications, and Systems*. Sept. 2007, pp. 1–8.

[383] J. Zuo, N. Ratha, and J. Connell. "Cancelable iris biometric." In: *19th International Conference on Pattern Recognition (ICPR 2008)*. Dec. 2008, pp. 1–4.

[384] M. Zviran and W. J. Haga. "Password security: an empirical study." In: *Journal of Management Information Systems* 15.4 (Mar. 1999), pp. 161–185.