JOHANNES KEPLER
UNIVERSITY LINZ

Enhancing Privacy-Preserving Biometric Authentication through Decentralization

Author
DI **Philipp Hofer**, BSc
01610705

Submission
**Institute of
Networks and Security**

Thesis Supervisor and
First Evaluator
Univ.-Prof. Dr.
René Mayrhofer

Second Evaluator
Prof. Dr. **Kristof Van Laerhoven**

Assistant Thesis
Supervisor
Dr. **Michael Roland**

September 2024

Doctoral Thesis

to confer the academic degree of

Doktor der Technischen Wissenschaften

in the Doctoral Program

Technische Wissenschaften

**JOHANNES KEPLER
UNIVERSITY LINZ**
Altenberger Straße 69
4040 Linz, Austria
jku.at

Abstract

This thesis explores the potential of decentralized technologies for enhancing privacy and operational efficiency within biometric authentication systems. The widespread use of centralized biometric systems is associated with significant risks, such as data breaches and privacy violations, highlighted by vulnerabilities in systems like India's Aadhaar. Promoting a shift towards decentralized frameworks, it allows users to control where their personal data is stored, aiming to reduce the risks of large-scale unauthorized access.

This research aims to enhance biometric systems for embedded devices through a holistic approach that progresses systematically from individual data elements, specifically embeddings, to complete application scenarios utilizing state-of-the-art technologies. The study begins by reducing the embedding size by 96 %, substantially increasing the processing efficiency of personal identifiers. Subsequently, the focus shifts to optimizing the most time-intensive component of the sensor by incorporating multiple face detection models that enhance specific operational efficiencies. Furthermore, developing a domain-specific sensor language allows for a precise definition of performance standards across various applications, facilitating a tailored and fully realized implementation that meets real-world requirements.

Testing a real-world prototype with cameras that incorporate the suggested improvements validates the effectiveness of decentralized biometric systems. This research demonstrates practical, efficient, and decentralized methods for authentication, making a significant contribution to the field and setting the stage for future studies in secure digital solutions focused on privacy.

Kurzfassung

In dieser Arbeit wird das Potenzial dezentraler Technologien zur Verbesserung des Datenschutzes und der Effizienz biometrischer Authentifizierungssysteme untersucht. Der weit verbreitete Einsatz zentralisierter biometrischer Systeme ist mit erheblichen Risiken verbunden, wie etwa Datenschutzverletzungen und Eingriffen in die Privatsphäre, die durch Schwachstellen in Systemen wie dem indischen Aadhaar-System deutlich werden. Durch die Verlagerung zu dezentralen Systemen können die Nutzer bestimmen, wo die persönlichen Daten gespeichert werden, um das Risiko eines unbefugten Zugriffs zu verringern.

Diese Forschung zielt darauf ab, biometrische Systeme für eingebettete Geräte durch einen ganzheitlichen Ansatz zu verbessern. Dabei wird systematisch von einzelnen Datenelementen bis hin zu vollständigen Anwendungsszenarien fortgeschritten und modernste Technologien genutzt. Die Studie beginnt mit einer 96-prozentigen Reduzierung der Größe von Embeddings, wodurch die Verarbeitungseffizienz von persönlichen Identifikatoren erheblich gesteigert wird. Anschließend wird der Schwerpunkt auf die Optimierung der zeitaufwendigsten Komponente des Sensors gelegt, indem mehrere Modelle zur Gesichtserkennung integriert werden. Die Entwicklung einer Sensorsprache ermöglicht die präzise Definition von biometrischen Anforderungen für verschiedene Anwendungen und erleichtert so eine maßgeschneiderte und vollständig realisierte Implementierung, die den realen Anforderungen gerecht wird.

Die Wirksamkeit dezentraler biometrischer Systeme wird durch einen Prototyp mit den vorgeschlagenen Verbesserungen validiert. Diese Forschungsarbeit demonstriert praktische, effiziente und dezentralisierte Methoden zur Authentifizierung und leistet einen bedeutenden Beitrag in diesem Bereich. Sie schafft die Voraussetzungen für künftige Studien zu sicheren digitalen Lösungen, die sich auf den Datenschutz konzentrieren.

Funding

This work has been carried out within the scope of Digidow, the Christian Doppler Laboratory for Private Digital Authentication in the Physical World. We gratefully acknowledge financial support by the Austrian Federal Ministry of Labour and Economy, the National Foundation for Research, Technology and Development, the Christian Doppler Research Association, 3 Banken IT GmbH, ekey biometric systems GmbH, Kepler Universitätsklinikum GmbH, NXP Semiconductors Austria GmbH & Co KG, and Österreichische Staatsdruckerei GmbH.

Acknowledgements

Just as a decentralized system relies on multiple nodes, this thesis owes its existence to a network of incredible individuals. At the core of this network was my advisor, René Mayrhofer. Your guidance was as precise and reliable as a well-tuned facial recognition algorithm. Our efficient meetings consistently recalibrated my focus and motivated me to push the boundaries of decentralized authentication. Thank you for being the secure backbone of my academic pursuits.

The Digidow team and my colleagues served as distributed nodes of support, contributing invaluable collaboration, fresh ideas, and unwavering encouragement. You have all left your fingerprints on this work, and while I am deeply grateful to everyone, I would like to highlight the contributions of two individuals: Michael Roland, who has an uncanny ability to spot the tiniest of errors—thank you for your meticulous and invaluable feedback. And Gerald Schoiber, for guiding me through the twists and turns of Rust and helping me navigate even the trickiest of ascents with a smile.

My family has been the trusted private key to my success, unlocking opportunities and helping me decrypt challenges throughout this academic journey. Your steadfast support has been the foundation of all my achievements.

To Marie: while I explored the intricacies of decentralized authentication, you were my constant, centralized source of support. Thank you for reminding me of the beauty beyond the realm of biometrics and decentralization, and for your endless patience in reviewing my work.

Completing this thesis has been a long journey, and I am both proud and relieved to reach this milestone, ready to embark on new endeavors with the knowledge and experience gained.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

In today's digital identity authentication landscape, biometric systems have become the preferred standard because they provide rapid, accurate, and user-friendly verification, surpassing traditional methods in security and convenience [151]. Traditional methods like passwords, PINs, and physical tokens often fall short in terms of security and convenience. Passwords can be forgotten or stolen, and physical tokens can be lost or duplicated. Biometric systems, which use unique biological traits such as fingerprints, facial features, or iris patterns, offer a more reliable solution by ensuring that the authentication factor is inherently tied to the individual. However, biometric authentication comes with its own set of problems, such as the potential for false positives/negatives, privacy concerns, the risk of data breaches, and the non-replaceability of compromised biometric data.

Centralized biometric systems have emerged as the predominant model for implementing these technologies. For instance, India's Aadhaar system exemplifies this model by consolidating biometric data, such as fingerprints and iris scans, in a central database, allowing for efficient data management and rapid authentication processes. Similarly, China's social credit system integrates biometric information to track and evaluate individuals' behavior, showcasing how centralization simplifies deployment and maintenance. By consolidating biometric data in a single location, organizations can efficiently manage large volumes of data and perform quick matches against a central database. This model simplifies the deployment and maintenance of biometric systems, making it easier to integrate into various applications, from mobile devices to national identification programs and even corporate security systems.

However, the convenience and efficiency of centralized biometric systems come with significant vulnerabilities that pose severe risks to privacy and security. The centralization of sensitive biometric data creates a single point of failure, making these systems highly susceptible to large-scale data breaches. High-profile incidents have demonstrated the catastrophic consequences of such breaches. Millions of individuals' personal data can be exposed, leading to identity theft, fraud, and other malicious activities.

Moreover, centralized biometric systems inherently facilitate a level of surveillance and control that conflicts with the principles of personal privacy and autonomy. The concentration of data in the hands of a few entities—whether governments, corporations, or other organizations—raises concerns about misuse and unauthorized access. This context sets the stage for exploring alternative approaches to biometric authentication that prioritize decentralization, aiming to mitigate the inherent risks of centralized models and champion user privacy and data sovereignty.

Addressing these challenges requires a shift in focus towards developing and optimizing biometric technologies within decentralized frameworks. This shift is motivated by two goals: enhancing the security and efficiency of biometric authentication processes and establish privacy-preserving mechanisms that are resilient against the pitfalls of centralized data storage. This effort is rooted in the understanding that the future of secure and convenient authentication lies in systems that not only recognize individuals with high accuracy but also respect and protect their privacy.

This thesis is part of the larger project Digidow[1] aimed at showing the possibility of biometric authentication in a decentralized way. While Digidow encompasses various aspects of decentralizing biometric systems, this thesis specifically focuses on making the sensor side more efficient. This research is dedicated to overcoming the limitations posed by embedded systems, focusing on creating scalable solutions that ensure robust authentication across a spectrum of technological contexts, without needing large-scale (and thus likely central) GPU clusters. The challenge lies in designing systems that are lightweight and efficient, yet capable of performing biometric recognition tasks without relying on centralized infrastructure.

In essence, this work addresses some of the fundamental flaws of centralized biometric authentication systems by advancing the field towards decentralized, privacy-centric models. By leveraging the principles of decentralization, this research aims to contribute to a new paradigm in biometric authentication—one that decentralizes data control, enhances security, and empowers users with greater autonomy over their personal information. The ultimate goal is to contribute to the development of a more secure, private, and user-centric digital identity verification system that aligns with our society's evolving needs and values.

1.2 Objectives

The overarching goal of our research is to design and implement an end-to-end system that enables the use of biometric authentication on embedded devices, meeting a set of key criteria:

1. **Decentralization:** The system should operate without reliance on centralized servers or entities for its core authentication functions. A wide range of infrastructures should be supported without leaning on centralized coordination or directories. A universal protocol should allow devices to discover and authenticate each other autonomously, enabling a network where any authorized registration can be seamlessly conducted. This protocol should

[1] https://digidow.eu

ensure that transmitting sensor data to the recognized individual does not hinge on specific operators or even device implementations but is universally applicable, thereby fostering a truly decentralized and inclusive authentication environment.

2. **Executable on embedded device (low latency / template generation):** Recognizing the impact of latency on user experience, the system aims for authentication processes to be completed swiftly to foster user acceptance. Matching the speed of contactless payments (300-500 ms) may be ambitious for a distributed system. However, the design will strive to minimize latency to levels comparable with or better than existing digital identity verification times, ideally keeping total transaction times well under the 30-second threshold deemed acceptable for activities like border checks.
3. **Feasibility:** The proposed system must be practical to implement with current technology, avoiding dependency on theoretical or unproven technologies. This feasibility ensures that the system can be developed, deployed, and evaluated in real-world conditions, facilitating empirical assessments of its effectiveness and areas for improvement.

By adhering to these objectives, our research to create a biometric authentication system that is not only technologically advanced and secure but also respectful of user privacy and practical for widespread adoption. This system will pave the way for a new paradigm in authentication technology, specifically tailored for the increasingly prevalent embedded devices in our digital ecosystem.

Non-Objectives

Within the confines of this thesis, the development or re-training of biometric models is intentionally set aside. This approach stems from a strategic choice to prioritize the integration of existing state-of-the-art biometric technologies. Such a decision is rooted in the desire for flexibility and adaptability. The biometric field evolves swiftly, with new advancements emerging regularly. Committing to a single, potentially quickly outdated model would detract from our system's ability to evolve. Furthermore, training biometric models demands significant computational resources and power. By sidestepping the creation and training of new models, this research maintains a sharp focus on its core aim: to design a decentralized, privacy-preserving authentication framework. It leaves the specialized domain of biometric model innovation to those with a primary focus in that field.

1.3 Approach

A thorough literature review is conducted to identify existing challenges in centralized biometric systems and the potential benefits of decentralization. The next step focuses on designing a decentralized framework for biometric authentication. This involves developing efficient algorithms for data capture, processing, and matching that can be executed on embedded devices. These algorithms are tested for performance and reliability in various conditions to meet the necessary benchmarks. To validate our research, a living lab is implemented as a real-world testing environment. This allows for iterative testing and refinement, ensuring the system is both theoretically sound and practically viable. The research integrates existing biometric technologies to maintain flexibility and to leverage state-of-the-art advancements. The overall goal is to create a scalable, privacy-preserving biometric sensor that can be implemented in real-world scenarios, providing a practical alternative to centralized models. Ultimately, all components contribute to the creation of a scalable, privacy-preserving biometric sensor, designed for implementation in real-world scenarios as a practical alternative to centralized models.

1.4 Contributions

This doctoral research presents advancements in decentralized biometric authentication systems, focusing on improving efficiency and applicability on embedded devices.

The research begins with optimizing biometric embeddings, achieving a 96 % reduction in size. This significant reduction directly addresses the challenges faced by decentralized systems, where limited processing power and storage capacity are typical constraints. The streamlined embeddings not only enhance the efficiency of these systems but also prove essential in multi-party computation scenarios, where multiple parties collaborate while keeping their inputs confidential. Moreover, in contexts where biometric data is stored on smartcards, the reduced size facilitates more efficient storage. This optimization also plays a critical role in minimizing the bandwidth required for data transmission, which is particularly important when, e.g., transferring embeddings within Tor introduction cells. By reducing the data footprint, the research enhances both the practicality and security of decentralized biometric systems.

Building upon this foundation, the research addresses the challenge of implementing facial recognition in resource-limited environments. Centralized systems often rely on large GPU clusters to manage the computational demands of facial recognition, particularly the face detection stage, which is the most time-consuming component of the recognition pipeline. I developed a multi-model face detection approach that allows to carefully balance the trade-off between processing speed and accuracy. Each model is optimized for specific facets of the task, ensuring that the system remains both reliable and efficient.

In addition, I developed a domain-specific sensor language designed to enhance the flexibility of biometric sensors. This language provides a standardized framework for defining and managing biometric requirements across different entities, ensuring that biometric data can be securely and efficiently processed.

The research culminated in the creation and deployment of a functional prototype, which involved the installation of three cameras in the hallway of our institute and an additional camera at a private residence in a different location.

This real-world implementation was key to assessing the practicality and effectiveness of the theoretical improvements explored in this thesis. By testing the system under everyday conditions, the prototype provided clear evidence of its performance and demonstrated the potential for future applications of decentralized biometric systems in real-world scenarios.

This research not only advances the state of decentralized biometric authentication but also lays a strong foundation for future innovations in secure, efficient, and adaptable biometric systems, particularly in environments with constrained resources and high privacy requirements.

1.5 Publications

Several sections of this thesis have been peer-reviewed and published in scientific workshops, conferences, and journals, with me as the main author:

1. **Hofer, Philipp**, Michael Roland, Philipp Schwarz, Martin Schwaighofer, and René Mayrhofer. 2021. Importance of different facial parts for face detection networks. In *2021 9th IEEE International Workshop on Biometrics and Forensics (IWBF)*. IEEE, Rome, Italy, (May 2021), pp. 1–6. DOI: 10.1109/IWBF50991.2021.9465087
2. **Hofer, Philipp**, Michael Roland, Philipp Schwarz, and René Mayrhofer. 2023. Efficient Aggregation of Face Embeddings for Decentralized Face Recognition Deployments. In *Proceedings of the 9th International Conference on Information Systems Security and Privacy (ICISSP 2023)*. SciTePress, Lisbon, Portugal, (February 2023), pp. 279–286. DOI: 10.5220/0011599300003405
3. **Hofer, Philipp**, Michael Roland, René Mayrhofer, and Philipp Schwarz. 2023. Optimizing Distributed Face Recognition Systems through Efficient Aggregation of Facial Embeddings. *Advances in Artificial Intelligence and Machine Learning*, 3, 1, (February 2023), 693–711. DOI: 10.54364/AAIML.2023.1146
4. **Hofer, Philipp**, Philipp Schwarz, Michael Roland, and René Mayrhofer. 2023. Face to Face with Efficiency: Real-Time Face Recognition Pipelines on Embedded Devices. In *21st International Conference on Advances in Mobile Computing & Multimedia Intelligence (MoMM 2023)*. ACM, Bali, Indonesia, (December 2023)
5. **Hofer, Philipp**, Philipp Schwarz, Michael Roland, and René Mayrhofer. 2024. Shrinking embeddings, not accuracy: Performance-Preserving Reduction of Facial Embeddings for Complex Face Verification Computations. In *14th International Conference on Pattern Recognition Systems (ICPRS 2024)*. IEEE, London, UK, (July 2024)
6. **Hofer, Philipp**, Philipp Schwarz, Michael Roland, and René Mayrhofer. 2024. BioDSSL: A Domain Specific Sensor Language for global, distributed, biometric identification systems. In *12th IEEE International Conference on Intelligent Systems (IEEE IS 2024)*. IEEE, Golden Sands, Bulgaria, (August 2024)

Moreover, I have also contributed to a few non-peer-reviewed publications and presentations, which contain results from this thesis:

1. **Hofer, Philipp**. 2022. Die Bedeutung verschiedener Gesichtsteile für Gesichtserkennung und dessen Zusammenführung. In *IKT-Sicherheitskonferenz 2022*. Vienna, Austria, (September 2022). https://www.digidow.eu/publications/2022-hofer-iktsicherheitskonferenz/Hofer_2022_IKTSicherheitskonferenz2022_Poster.pdf
2. **Hofer, Philipp**, Michael Roland, Philipp Schwarz, and René Mayrhofer. 2022. Efficient aggregation of face embeddings for decentralized face recognition deployments (extended version). (December 2022). https://arxiv.org/abs/2212.10108
3. **Hofer, Philipp**. 2023. Dezentrale Gesichtserkennung. *OCG Journal*, 48, 1, (April 2023), 14–15. https://www.ocg.at/sites/ocg.at/files/medien/pdfs/OJ2023-01.pdf

Additionally, the following publications, where I contributed as a co-author, have been published. These works are relevant to this thesis as they explore gait recognition, which complements the biometric methods discussed in this research, contributing to the broader understanding of biometric systems and their applications:

1. Philipp Schwarz, Josef Scharinger, and **Hofer, Philipp**. 2021. Gait recognition with densePose energy images. In *International Conference on Systems, Signals and Image Processing*. Springer, pp. 65–70
2. Philipp Schwarz, **Hofer, Philipp**, and Josef Scharinger. 2022. Gait Recognition Using 3D View-Transformation Model. In *International Conference on Computer Aided Systems Theory*. Springer, pp. 452–459

In addition to these contributions, I am pleased to report that NDR reached out to me for an interview on face recognition, prompted by the impact of my recent publications. The resulting documentary is scheduled for release in September 2024. Furthermore, although a side project, I was awarded a rectorate bonus for my experiments in voice cloning, which, while tangential, contributed to the broader interdisciplinary nature of this research.

1.6 Outline

The structure and flow of the thesis are visually summarized in Figure 1.1. The journey begins with an overview of existing research and foundational concepts in Chapter 2 “Background”. This sets the stage by reviewing the state-of-the-art in biometric systems and highlighting the motivations for shifting towards decentralized models.

Building on this foundation, “Understanding facial features in biometric authentication” (Chapter 3), examines the components that constitute an embedding and focusing on the significance of individual facial parts.

Following this, the discussion advances to “Shrinking giants: The power of tiny embeddings” (Chapter 4), where strategies for making single embeddings more efficient are explored. This chapter examines methods to reduce the size of embeddings while maintaining their performance, thereby contributing to the overall efficiency of the biometric system.

With these optimized embeddings in place, the focus shifts in “One template to rule them all: Fusing embeddings” (Chapter 5) to combine multiple embeddings to create a comprehensive and accurate representation of an individual. This chapter details the process of merging embeddings to form templates that represent a person effectively.

Having established a robust template, the thesis then explores “The speed of sight: Optimizing face detection for embedded systems” (Chapter 6), investigating the implementation of an efficient pipeline on embedded systems. This chapter covers integrating and optimizing the template generation process on hardware with limited resources, ensuring its practical applicability.

Next, “Biometric Domain Specific Sensor Language (BioDSSL)” (Chapter 7) introduces an application framework that allows for the specification and management of different biometric modalities. This chapter discusses the design and implementation of BioDSSL, which enhances the flexibility and efficiency of integrating multiple sensors and modalities.

Finally, all the components developed in the previous chapters are integrated into a prototype. In Chapter 8 “When Theory Hits Reality: Living lab prototype and Digidow integration”), we evaluate the prototype in real-world scenarios to assess its performance, scalability, and potential for practical deployment.

The last Chapter, “Conclusion and outlook”, summarizes the essential findings and contributions of the research. It discusses potential future directions and broader implications for the field of biometric authentication and privacy-preserving technologies.

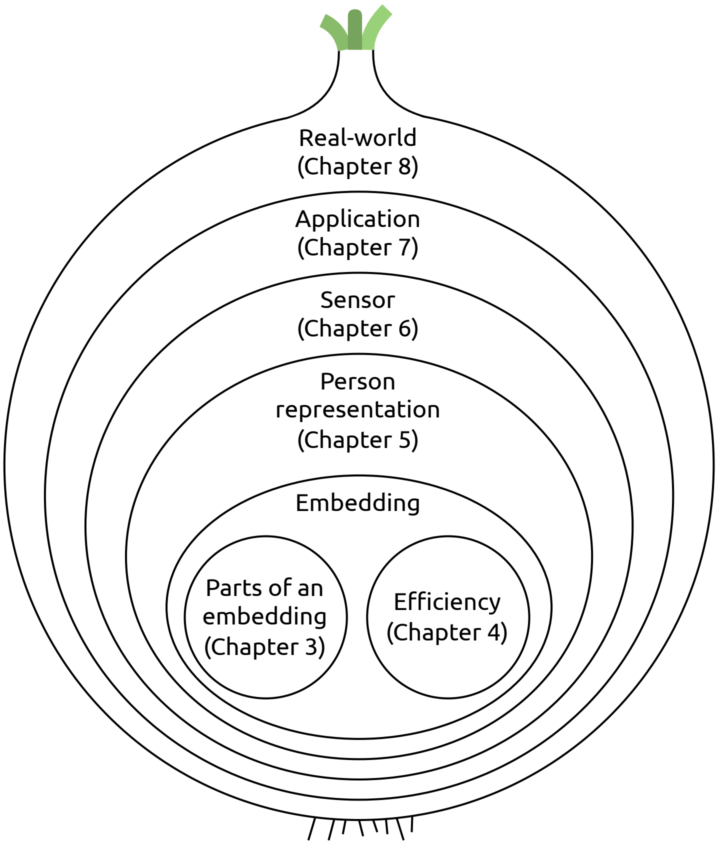


Figure 1.1: This image illustrates the progressive layers of my research, starting with foundational improvements in embeddings, followed by creating a comprehensive single-person representation through integrating multiple embeddings. The next layer represents the development of a sensor based on this representation. Subsequently, this sensor is integrated into a practical application, and finally, the outermost layer demonstrates the application's real-world impact and effectiveness. Each layer builds upon the previous, symbolizing the cumulative advancement of knowledge and technology throughout the thesis.

Chapter 2

Background

In this chapter, we provide an overview of the context and prior work relevant to this thesis. We begin with a review the various components and methods involved in biometric authentication (Section 2.1). Finally, we introduce and detail the datasets utilized in this thesis (Section 2.2).

2.1 Decentral biometric authentication

Decentralized biometric authentication systems offer substantial potential for enhancing security and privacy. This chapter examines the shift from centralized to decentralized architectures in biometric systems.

We review centralized biometric systems, drawing on existing research to outline their key methodologies and inherent challenges. This review sets the stage for understanding the motivations behind moving toward decentralized systems. These motivations include the need for enhanced privacy, reducing the risks of large-scale data breaches, and empowering users with control over their biometric information.

We briefly overview various biometric modalities, directing readers to surveys on these topics, but our primary emphasis is on how decentralization impacts the design and functionality of biometric systems. We explore the parts of these systems, from hardware to algorithms, and discuss how they integrate with broader technologies like internet of things and mobile devices.

The unique security and privacy challenges of decentralized systems and strategies for managing these issues are also examined.

The integration of biometric authentication into modern security frameworks marks a notable evolution in identity verification and access control. The rationale for adopting biometric methods over traditional authentication mechanisms, such as PINs, passwords, or smart cards, is underpinned by a combination of convenience, enhanced security, and user-centric considerations:

- Convenience and user experience: Traditional authentication methods can be cumbersome for users, such as remembering (hopefully complex) passwords or carrying physical tokens like smart cards or FIDO keys. Biometric authentication, leveraging inherent traits like fingerprints or facial recognition, offers a more seamless experience. However, there is a degree of user hesitancy toward biometrics, often due to privacy concerns [6]. Despite this, the approach simplifies the process by mitigating issues like forgotten passwords or lost tokens, provided that robust data protection measures and clear privacy policies as well as alternative recovery procedures and options for users without these biometrics are in place to address these apprehensions.

- Enhanced security: Biometric authentication can provide a higher security level than traditional methods. Passwords and PINs, which can be shared, guessed, or stolen, represent a weaker link in security chains. The intrinsic link between the individual and their biometric traits theoretically ensures that only authorized persons gain access, minimizing the risks associated with compromised credentials. However, it is important to recognize that *ensuring* might be too strong a term, as false positives, albeit uncommon, can occur in biometric systems.
- Non-transferability and accountability: Unlike passwords or smart cards, biometric traits are inherently non-transferable. This characteristic ensures that access rights cannot be easily transferred or shared, leading to greater accountability in transactions and interactions. The non-transferability of biometrics instills a higher degree of trust in the authentication process, which is particularly crucial in sensitive applications.

However, while the primary motivation for implementing biometric authentication systems is to enhance security and convenience for authenticated individuals, these systems are often not primarily designed with the technical safeguards necessary to reduce the potential for surveillance or invasive monitoring. These concerns stem from the inherent design of such systems, which often prioritize security and convenience without sufficient emphasis on privacy protection. The ethical implementation and use of biometric authentication must, therefore, be underpinned by stringent privacy and data protection standards. This necessity highlights the importance of exploring alternative system architectures, including technical solutions beyond traditional legal safeguards, to ensure the technology aligns with user privacy concerns and fundamental rights. Exploring these alternative architectures and their implications for privacy and data control will be further elaborated in Section 2.1.

To summarize, the shift toward biometric authentication is driven by its ability to offer a more secure, convenient, and user-friendly alternative to traditional methods. This transition, however, must be navigated with careful consideration of ethical implications and privacy concerns, particularly when personal and sensitive biometric data is involved.

Motivation for decentral architectures

The pervasive adoption of biometric authentication technologies across various sectors underscores the necessity to critically evaluate the underlying data management models, especially the predominant centralized systems. Despite their operational efficiencies, centralized models are riddled with inherent vulnerabilities that significantly compromise user privacy and data security.

Centralized biometric systems, as implemented in countries like China [121], India [212], Russia [140], and as proposed in the European Union [58], consolidate biometric data from millions, sometimes billions, of individuals into a single, centralized database. This aggregation of sensitive data introduces several downsides:

- Lack of user control: In centralized systems, users often have limited control over their data post-collection, leading to significant privacy concerns. However, not all systems inherently lack user control. For instance, Austria's ELGA system[1] allows participants to view their health records, check who accessed their data and when, and modify legal access permissions. Despite this, users must still rely on the system's integrity and compliance and ultimately trust the operator to respect these settings as they cannot independently verify compliance.
- Heightened vulnerability to data breaches: Centralized repositories of biometric data are lucrative attack targets. The consolidated nature of these systems means that a successful breach can compromise an extraordinarily large volume of sensitive data. The risk to centralized systems is far from hypothetical; numerous incidents have highlighted their vulnerability to a range of attacks. High-profile data breaches have spanned multiple sectors and companies, impacting not only mainstream technology firms such as Google[2], Microsoft[3], Facebook[4], Uber[5], and Twitter/X[6], but also entities specializing in security, like Lastpass[7]. Furthermore, organizations that handle highly sensitive personal data, such as 23andme[8] and Ancestry[9], have been compromised. Additionally, government systems in countries including Indonesia[10], the US[11], and the UK[12] have also been breached, underscoring the pervasive nature of this issue. Furthermore, the Aadhaar database, containing biometric and identity information on over 1.1 billion Indian citizens, experienced a significant breach that exposed names, unique identity numbers, and bank details[13]. More known data breaches are visualized e.g. at https://informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/ and https://en.wikipedia.org/wiki/List_of_data_breaches.

[1] https://www.elga.gv.at/

[2] https://www.theverge.com/2018/12/10/18134541/google-plus-privacy-api-data-leak-developers

[3] https://www.forbes.com/sites/daveywinder/2020/01/22/microsoft-security-shocker-as-250-million-customer-records-exposed-online

[4] https://www.businessinsider.com/stolen-data-of-533-million-facebook-users-leaked-online-2021-4

[5] https://restoreprivacy.com/uber-data-leak-breach-third-party-vendor-hacked/

[6] https://firewalltimes.com/twitter-data-breach-timeline/

[7] https://techcrunch.com/2022/12/14/parsing-lastpass-august-data-breach-notice/

[8] https://arstechnica.com/tech-policy/2023/12/hackers-stole-ancestry-data-of-6-9-million-users-23andme-finally-confirmed/

[9] https://arstechnica.com/tech-policy/2023/12/hackers-stole-ancestry-data-of-6-9-million-users-23andme-finally-confirmed/

[10] https://kr-asia.com/shoddy-data-protection-in-indonesia-threatens-personal-security-of-citizens

[11] https://www.nytimes.com/2016/02/09/us/hackers-access-employee-records-at-justice-and-homeland-security-depts.html

[12] https://www.thecricketer.com/Topics/grassroots/ecb_issue_warning_to_users_of_online_coaching_platform_following_data_breach.html

[13] https://www.zdnet.com/article/another-data-leak-hits-india-aadhaar-biometric-database/

- Surveillance and privacy intrusions: Centralization of biometric data lends itself to potential abuse for surveillance and other privacy-invasive practices. The concentration of data in a single entity's hands increases the risk of unauthorized access and misuse, particularly by state actors or rogue elements within organizations.
- Systemic and operational risks: Centralized systems, due to their scale and complexity, are prone to operational risks such as system failures, downtime, or bottlenecks. The failure of a centralized system can have widespread implications, affecting large numbers of users simultaneously.

In contrast, a decentralized approach to biometric authentication addresses these concerns by dispersing data across multiple nodes. This model inherently dilutes the value of any single point of attack, thereby reducing the incentive for large-scale breaches. Decentralization also enhances user privacy and control, as biometric data is not wholly reliant on a single entity's policies or security measures. It supports a more democratic and user-centric model of data governance, where users have a greater say and visibility over their data.

However, the transition to decentralization is accompanied by its own set of challenges. Decentralized systems are inherently complex, requiring robust coordination and security protocols to ensure data integrity and system reliability. The balance between enhanced privacy and security on one side and increased system complexity on the other is a central consideration in the development of decentralized biometric authentication systems. Additionally, the scattered nature of decentralized systems could result in reduced availability.

To summarize, while centralized biometric authentication systems present operational efficiencies, their significant drawbacks in terms of data security, privacy risks, user control, and systemic vulnerabilities motivate the exploration of decentralized models. These models promise enhanced security and privacy but require careful design to manage their inherent complexities. Therefore, the integration of biometrics with decentralized platforms is emerging as a new frontier, aiming to address privacy and security concerns in digital identity management [132].

2.1.1 Definitions

This subsection aims to clarify the terms used in this thesis.

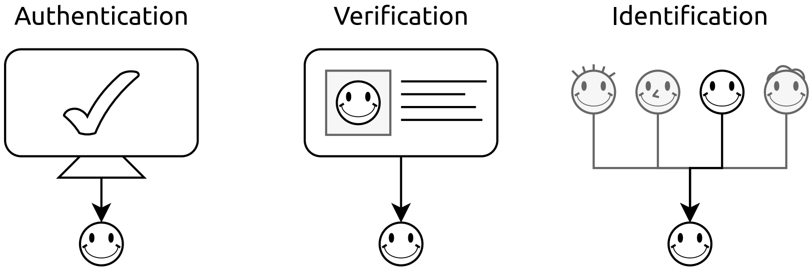


Figure 2.1: Visualization of differences between authentication, verification, and identification.

- Biometric authentication, verification, and identification (Fig. 2.2)
  - **Biometric Authentication:** A broad term that involves using unique biological characteristics to verify an individual's identity. This process typically compares biometric data, such as fingerprints or facial features, against a stored profile to ensure that access is granted only to authorized users.
  - **Biometric Verification:** Also known as 1:1 comparison, biometric verification involves comparing an individual's biometric data against a specific template to confirm their claimed identity. This method ensures that individuals are who they claim to be by matching their biometric traits, such as fingerprints or facial patterns, against a previously enrolled template.
  - **Biometric Identification:** This process involves recognizing individuals based on their unique physical or behavioral characteristics through a one-to-many (1:N) comparison. It scans a database to find a match for the presented biometric data, identifying an individual among many without any prior claim of identity.

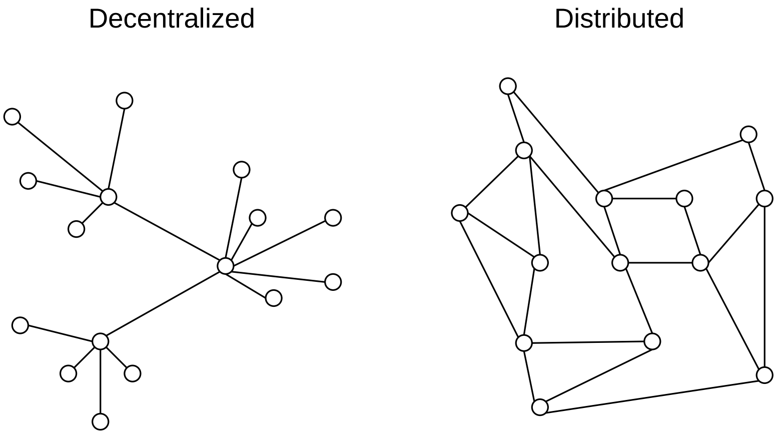


Figure 2.2: Illustration highlighting the key distinctions between decentralized and distributed systems.

- Decentralized and distributed systems
  - **Decentralized:** In a decentralized system, control is distributed among multiple entities or nodes, with no central authority. This structure enhances resilience and reduces single points of failure as decision-making and operations are spread across the network.
  - **Distributed:** A distributed system consists of multiple parts located on different networked computers that communicate and coordinate their actions by passing messages to achieve a common goal. Despite the physical distribution of resources, these systems are typically controlled by a single logical entity, ensuring coherent operation and management while improving scalability and fault tolerance.

2.1.2 Centralized biometric authentication

While the exploration of emergent decentralized biometric authentication systems is in its infancy, a substantial corpus of research exists centered on traditional centralized biometric authentication systems. This existing body of literature offers significant insights into the design and implementation of biometric systems, insights that remain relevant to the development of decentralized counterparts despite their architectural divergence.

1. Surveys on biometric modalities: The focus is on specifics about modalities.
    - Rui et al. [173] first defines requirements (accuracy, efficiency, usability, security, and privacy) and then evaluates various modalities based on them.
    - Karimian et al. [104] focused on ECG and iris signals.
    - Lien et al. [118] focus on details about many modalities.
2. Central system architecture proposals: The proposed architectures assume a single logical entity.
    - Karimian et al. [104] discuss the (hardware) setup of a biometric system, where all components are inside a single logical component (a mixture between our proposed 3 components).
3. Focus on specific challenges
    - Carmel et al. [28] focuses on identity theft prevention.
    - Sundararajan et al. [203] and Basco et al. [19] focuses on wearable modalities. The former highlights the differences in modality characteristics, system performance, and security considerations. The latter focuses on the unique challenges and computational considerations inherent in wearable biometrics, highlighting the differences between traditional systems in sensor constraints, data processing, and authentication mechanisms.
    - Blanco-Gonzalo et al. [18] focuses on the usability of biometric systems, which impacts system efficacy and user acceptance.
    - Wong-In et al. [225] focuses on applying biometric verification on educational examiners.
    - Song et al. [196] focuses on feature-level fusion of facial and acoustic features.

In this chapter, we focus on the complexities of emergent decentralized biometric authentication systems as a cohesive network of intercommunicating components. This perspective is important for several reasons:

1. **Comprehensive coverage:** Unlike previous surveys that focus on the specifics of biometric modalities or the architectures of centralized systems, in this chapter, we cast a wider net, acknowledging the intricate web of interactions within decentralized systems.

2. **Decentralization and system components:** In this chapter, we have an explicit focus on the structural decentralization of biometric systems. By decomposing the system into distinct but interconnected components, this chapter serves as a guide for building decentralized architectures.
3. **Strategic emphasis on architecture over modality:** While recognizing the importance of modality-specific challenges and advances, this chapter intentionally shifts the focus toward the overarching architecture of decentralized systems. It treats modalities as integral yet subservient elements within a broader system architecture, thus offering an innovative perspective that prioritizes systemic coherence over modality-centric innovations. This approach does not diminish the value of modality research; instead, it contextualizes it within a larger framework, where the synergy between components and their communication protocols becomes a critical area of study.

In doing so, this chapter addresses a gap in the literature by emphasizing the "system of systems" approach to decentralized biometric authentication. This focus illuminates the unique challenges and opportunities presented by decentralized architectures—challenges that are often obscured when the lens is narrowly focused on individual components or centralized frameworks. By drawing attention to the dynamics of system-wide communication and integration, this work aims to pave the way for more resilient, efficient, and user-centric biometric authentication systems. Moreover, this chapter hopes to be a valuable resource for researchers and practitioners alike by selectively referring to specialized surveys on modality specifics and system components.

2.1.3 Biometric authentication overview

The components of biometric systems are shown in Figure 2.3. This figure aims to give a clear overview of all the possible components, showing the structure and how different parts of biometric systems relate to each other. Some parts, like the validity check and pre-selection, can be optional. Also, it is possible to do some processes concurrently, such as running the validity check and preprocessing together. This approach helps to understand the complexity and flexibility of biometric systems' setup and operation.

In centralized systems, these components are typically managed and controlled by a single entity, such as a specific company or governmental body. This entity typically operates its own infrastructure, encompassing e.g. facial recognition cameras, a database of employee or citizen information, and a matching system, all consolidated on a single logical unit of hardware. This integrated approach negates the necessity for compartmentalizing biometric systems, maintaining simplicity by centralizing everything on one platform.

Conversely, decentralized systems scatter these elements across various entities or locations, resulting in a more intricate system architecture. Such a configuration underlines the potential advantage of organizing these components into self-sufficient operational clusters. These clusters are capable of functioning independently and may extend beyond the boundaries of a singular administrative domain.

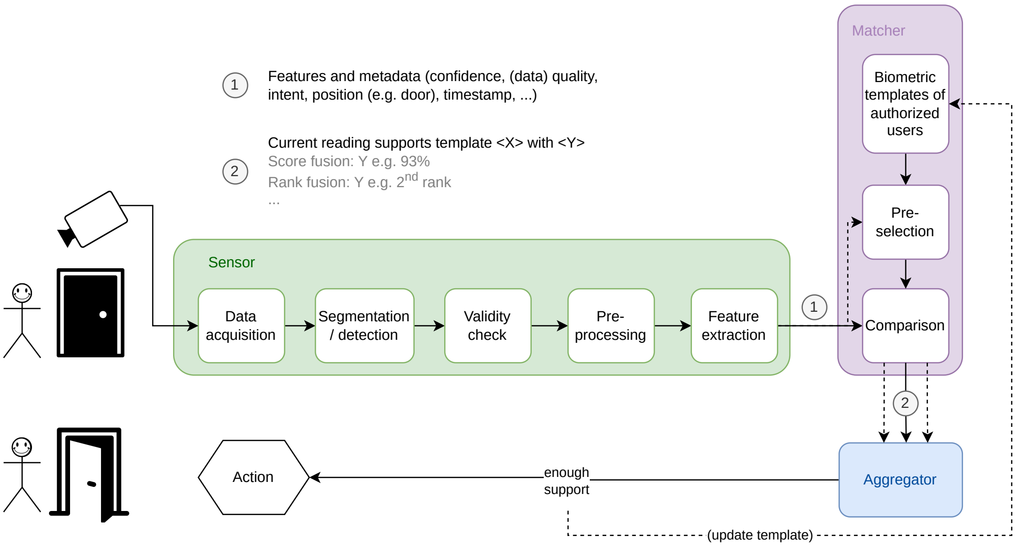

Matcher
1
Features and metadata (confidence, (data) quality, intent, position (e.g. door), timestamp, ...)
2
Current reading supports template <X> with <Y>
Score fusion: Y e.g. 93%
Rank fusion: Y e.g. 2nd rank
...
Biometric templates of authorized users
Pre-selection
Sensor
Data acquisition
Segmentation / detection
Validity check
Pre-processing
Feature extraction
1
Comparison
2
Action
enough support
Aggregator
(update template)

Figure 2.3: An overview of the components within a biometric system, illustrating the interaction and flow between the Sensor, Matcher, and Aggregator components. The system starts with data acquisition from a sensor, followed by segmentation, detection, and feature extraction. The Matcher compares extracted features with stored biometric templates, with potential pre-selection and validity checks. The Aggregator compiles the results to decide on the required action, highlighting the modularity and flexibility inherent in both centralized and decentralized biometric systems.

The key components of such biometric systems include Sensor, Matcher, and Aggregator. We divided the architecture into these components because it enables specialization and scalability within each phase of the authentication process, ensuring both flexibility and efficiency in handling diverse biometric inputs and authentication scenarios.

- **Split between Sensor and Matcher:** The rationale for separating the Sensor from the Matcher is fundamentally about enhancing security and operational flexibility in decentralized systems. The comparison of two features necessitates access to both the current live embedding (obtained by the sensor) and the stored template (in the matcher). However, in decentralized systems, the database containing the enrolled users' template might not be centrally located or directly accessible to the sensor. This is often a deliberate choice to prevent the leakage of sensitive information to the sensor. By segregating the Sensor and Matcher, we ensure that live biometric data and stored biometric templates remain isolated, reducing the risk of exposing personally identifiable information to vulnerabilities inherent in less secure environments. This separation allows for a safer handling of biometric data. The sensor focuses on capturing and preprocessing data, while the matcher securely performs comparisons without direct access to live data.
- **Split between Matcher and Aggregator:** The division between the Matcher and the Aggregator is driven by the need for a customizable and multimodal authentication framework. The Matcher's role is to compare the in-

coming biometric features against a database of authorized users' templates and generate a support level for template matches. The Aggregator then takes this support information and decides whether it meets the threshold for a particular action. This separation enables an entity to specify the exact level of authentication support required for different actions, enhancing security measures. For instance, more sensitive actions can be configured to require higher levels of support or authentication from multiple biometric modalities. This multi-modal approach, facilitated by the Aggregator, significantly enhances security by requiring multiple independent biometric verifications to agree before granting access or performing an action. Additionally, this structure allows for easier integration of future biometric technologies and modalities, as they can be added to the system with minimal changes to the existing infrastructure.

In summary, the separation into Sensor, Matcher, and Aggregator components is a strategic architecture choice that enhances the biometric system's security, flexibility, and scalability. It allows for the specialized handling of sensitive biometric data, customizable authentication thresholds, and the incorporation of multi-modal biometric inputs, thereby providing a robust framework for secure and efficient biometric authentication. The subsequent subsections will focus on each of these components in more detail, exploring the technologies, methodologies, and security measures underpinning this biometric authentication approach.

Historical perspective and evolution of biometric technologies

Early Beginnings: The historical roots of biometrics can be traced back to ancient civilizations, where physical characteristics like fingerprints were used for identification purposes. However, the formal application of biometrics began in the late 19th century. Alphonse Bertillon, a French police officer and biometrics researcher, pioneered the use of anthropometry—systematic measurements of body dimensions—for criminal identification [128].

Fingerprint Era: The early 20th century saw the rise of fingerprint identification as a reliable method. This period was marked by the development of systematic approaches for fingerprint classification and matching [16, 36].

Technological Advancements: The latter half of the 20th century witnessed significant technological breakthroughs. Automated fingerprint identification systems (AFIS) [14] were introduced, revolutionizing how fingerprints were processed and matched. This period also saw the advent of other biometric modalities such as facial recognition [70], iris scanning [49], and speaker recognition [190], fueled by advancements in digital imaging and computing power.

[14] https://en.wikipedia.org/wiki/Automated_fingerprint_identification

Integration and Expansion: The turn of the 21st century marked a phase of rapid integration and expansion of biometric technologies. In response to the heightened security concerns following the events of 9/11, biometrics found its way into various domains, from border control [139] to consumer electronics. The advent of machine learning and artificial intelligence further propelled this expansion, enhancing the accuracy and efficiency of biometric systems.

Current Trends and Future Outlook: Today, biometrics technology is part of security and identification systems worldwide. The focus has shifted towards developing multimodal biometric systems that use multiple biometric indicators, increasing reliability and preventing spoofing [73, 97, 112, 174, 184, 226, 246, 247].

2.1.4 Sensor

The sensor interacts with its environment through various means including RGB [33, 127, 146] / 3D [3] cameras, fingerprint readers [96, 131, 152, 160], infrared signals [2, 7, 206, 208], and microphones [21, 38, 114, 122]. Its core objective is to detect humans in its surroundings, and if someone is detected, **features** should be extracted, which can be used to identify a person. Given the human body's diversity in unique identifiers, biometric authentication systems exploit these through various modalities, each with its distinct set of benefits and challenges regarding accuracy, security, and user experience. Understanding the specific requirements of a modality allows for a tailored approach that optimizes the balance between these factors. This consideration ensures that the chosen biometric modality aligns with the desired level of security, meets accuracy expectations, and delivers a seamless user experience. Essential to this consideration are the following criteria:

- Inherent qualities of the modality
  - Universality: Every individual should possess this trait [44, 57, 108, 173, 202].
  - Uniqueness: The trait must be distinct for each individual, ensuring high accuracy in identification [57, 69, 108, 173, 202].
  - Permanence: The stability of the trait over time, which affects its reliability [20, 44, 57, 69, 108, 173, 202, 227].
- Technical and performance considerations
  - Collectability [57, 108, 202, 227]
    - Ease of Acquisition: How easily the biometric data can be captured (e.g., ears are visible, not generally obscured by makeup or sunglasses, and suitable for cheap sensor technology) [20, 44, 69, 173].
    - Operational Efficiency: Factors like distance from the sensor and environmental effects on data capture.
  - Performance [108]

    - Speed: The time required to process the biometric data [44, 57].
    - Computing Requirements: The computational power needed for data processing [44].
    - Verification Precision: The accuracy with which a system can confirm the identity of an individual [44, 57].
- Socio-technical aspects
  - Acceptability: The degree to which individuals are willing to use the biometric system, influenced by cultural and personal concerns [44, 57, 108, 173, 202].
  - Circumvention Difficulty: Specific to modalities like ear recognition, which are difficult to replicate through means like plastic surgery [20, 108, 202].

Many biometric modalities can fulfill these criteria and can be categorized based on

- Physiological or behavioral traits: Physiological traits include physical characteristics like fingerprints or iris patterns, while behavioral traits encompass actions such as typing rhythms or walking gait [20, 205],
- Soft or hard biometrics: Hard biometrics involve unique physical identifiers like DNA or retinal scans, whereas soft biometrics use less distinctive features like height or hair color [43, 98, 99, 147], and
- Human body class: This categorization refers to the particular body parts used for identification, such as the heart's ECG patterns or the face [167].

Table 2.1: Biometric Methods and their Characteristics

Method	**Description**	**Ref.**	**Physiol. / behavioral**	**Human body class**	**Soft vs hard**
Visible face rec	Utilizes unique facial features for identification using RGB sensors	[102]	phys.	face	hard
IR face rec	Utilizes unique facial features for identification using infrared sensors	[102]	phys.	face	hard
Iris	Identifies individuals based on unique patterns in the iris	[102]	phys.	ocular	hard

Method	Description	Ref.	Physiol. / behavioral	Human body class	Soft vs hard
Retina	Utilizes the unique patterns of the blood vessels at the back of the eye for identification	[102]	phys.	ocular	hard
Sclera	Identifies individuals based on the unique patterns of the white part of the eye	[102]	phys.	ocular	hard
Periocular	Uses the region surrounding the eye for identification, including the eyebrows and skin texture	[102]	phys.	ocular	hard
Fingerprint	Employs the unique patterns of ridges and valleys on a person's finger	[102]	phys.	palm and finger	hard
Palm	Uses the distinct lines and features of the palm	[102]	phys.	palm and finger	hard
Finger vein	Identifies individuals through the vein patterns in the fingers	[102, 178]	phys.	palm and finger	hard
Ear	Involves the distinctive shape and structure of the ear	[174]	phys.	face	hard
Blood vessel dynamics	Focuses on the unique patterns of blood vessels	[174]	beh.	vascular	hard
Photo-plethysmograph	Measures volumetric changes in blood flow, possible with a flashlight and camera or smartwatches, and is an emerging method	[155]	phys.	vascular	hard
Dental	Identifies individuals based on the unique structure and alignment of teeth	[29]	phys.	oral	hard

Method	**Description**	**Ref.**	**Physiol. / behavioral**	**Human body class**	**Soft vs hard**
Gait	Identifies individuals by their walking patterns	[187, 218]	beh.	mainly lower limbs	hard
Eye movement	Tracks unique eye movement patterns such as saccades and fixations	[102]	beh.	ocular	hard
ECG	Uses the unique electrical activity of the heart	[102, 118, 214]	beh.	Heart: electro-physio-logical	hard
ICG	Uses IR to recognice vascular anatomy	[102]	phys.	Heart: electro-physio-logical	hard
PCG	Identifies individuals based on phonocardiographic patterns, analyzing the heart's acoustic signals	[102]	phys.	Heart: acoustic	hard
Echocardiography	Uses ultrasound imaging to analyze the heart's structure and function for identification	[102]	phys.	Heart: structural	hard
SCG	Utilizes seismocardiographic signals to analyze the mechanical activity of the heart	[102]	phys.	Heart: mechanical	hard
BCG	Analyzes the ballistocardiographic signals representing the body's response to the ejection of blood by the heart	[102]	phys.	Heart: mechanical	hard
CM	Identifies individuals using cardiomechanical patterns, analyzing the movement of the heart	[102]	phys.	Heart: mechanical	hard

Method	Description	Ref.	Physiol. / behavioral	Human body class	Soft vs hard
PPG	Uses photoplethysmography to measure blood volume changes in the microvascular bed for identification	[102, 118]	phys.	vascular	hard
Voice	Relies on vocal characteristics for identification	[118]	beh.	vocal	hard
Hand gestures	Employs distinct hand movements and gestures	[66, 202]	beh.	upper limbs	hard
Signature	Identifies individuals based on the unique style and patterns of their handwriting	[118]	beh.	upper limbs	hard
Keystroke dynamics	Analyzes the rhythm and speed of a person's typing patterns for identification	[118]	beh.	upper limbs	hard
Breath	Unique characteristics of breathing patterns, based on audio or motion	[118]	beh.	respiratory	hard
Finger and mouse movements	Identifies individuals based on the unique characteristics of their finger and mouse movements	[24, 239]	beh.	upper limbs	hard
Skull conduct	Uses the unique way sound waves propagate through a person's skull for identification	[102]	beh.	brain	hard
EEG	Identifies individuals based on the unique patterns of electrical activity in their brain	[102]	beh.	brain	hard
Skin color	Identifies individuals based on their unique skin color or tone	[161]	phys.	skin	soft

Method	Description	Ref.	Physiol. / behavioral	Human body class	Soft vs hard
Clothing color	Identifies individuals based on the color of their clothing, often used as a supplementary feature	[161]	beh.	external appearance	soft

The discussion leads to the question: What constitutes features in this context? State-of-the-art techniques for feature extraction involve generating embeddings—numerical vectors that encapsulate an entity's distinctive attributes. Often generated through neural networks, these vectors are designed to maximize inter-class variation and minimize intra-class variation, utilizing a predetermined distance metric. This approach contrasts with traditional methods like fingerprint recognition, which compares minutiae points, showcasing the breadth of methodologies in feature extraction within biometric systems.

In addition to extracting features, the sensor is capable of detecting not only the presence of individuals (along with certain features for identification purposes) but also additional **metadata**. This metadata varies significantly based on the application and depends on what action should be executed. Examples of commonly collected metadata include:

- Timestamp: The exact time when the data was captured or the event occurred.
- Intent: The sensor can differentiate if an individual intends to perform a specific action, such as opening a door, or if they are engaging in everyday activities like enjoying a cup of coffee or conversing with a colleague nearby.
- Positional information: Where the individual is located relative to the sensor or an object of interest.
- Confidence: The sensor's self-assessed reliability in its readings, which can include quality metrics of the input data, such as evaluations of the data's integrity and usefulness for processing.

Further metadata examples to consider include the individual's velocity, the duration of their presence within the sensor's range, or environmental conditions like lighting and temperature, which could influence the sensor's performance or the interpretation of its readings.

Data acquisition

This initial step involves capturing raw biometric data through various specialized devices:

- RGB Cameras: Face recognition [137], iris recognition, gait recognition, hand [157] (-gesture [66, 138])

- Fingerprint Scanner/Reader: Fingerprint recognition [37, 77, 131, 150, 204, 237]
- Microphone: Voice/speaker recognition [170, 172, 201, 226, 234]
- Photoacoustic tomography [240]: combination of ultrasound and optical measurements
- Accelerometer [202]: gait and gestures recognition
- Wearable Fitness Trackers and Smartwatches [46, 56, 129, 207, 216]
- Thermal Imaging Cameras Heat signature recognition: face [8, 12, 180, 200], fingerprint [101], hand [41]

Segmentation / Detection

After the initial data acquisition, this phase involves isolating the biometric features of interest from the raw input. The techniques and challenges involved are specific to the biometric modality being processed. To avoid repetition, we will focus on discussing specific techniques for a few popular modalities listed in the table. This approach does not restrict our findings, as similar principles and techniques apply to the other modalities not covered in detail. Examples of segmentation and detection across various modalities include:

- **Face recognition:** Algorithms aim to identify and isolate the face from its surroundings using face detection, accounting for factors like diverse backgrounds, lighting variations, and the presence of multiple faces [74, 110, 238].
- **Fingerprint recognition:** The process distinguishes the fingerprint from the background, which is essential for accurate ridge and valley identification [26, 32, 59, 177, 219].
- **Iris recognition:** Detects the iris within an eye image, segmenting it from the pupil, sclera, and eyelids, despite the complexity of the eye structure [11, 116, 220, 222].
- **Gait recognition:** Identifies and segments the sequence of human movement or posture that makes up an individual's gait, which is challenging due to environmental variations and potential obstructions [55, 143, 228].
- **Hand gesture recognition:** Focuses on detecting and segmenting hand positions and movements from the background, recognizing hand shapes and finger configurations for gesture interpretation [25, 103, 144, 211, 249].
- **Voice/Speaker recognition:** In audio modalities, it involves isolating speech from background noise and detecting the start and end of spoken phrases, crucial for effective voice recognition [10, 22, 23, 135].

The segmentation and detection step ensures that only pertinent biometric data is forwarded through the authentication system.

Validity / Liveness check

The validity check is designed to assess the authenticity of the biometric data, ensuring that it represents a live and genuine subject rather than an artificial or fraudulent attempt to mimic biometric characteristics. This stage employs sophisticated methods to distinguish between legitimate biometric samples and potential spoofs, artifacts, or noise, thus bolstering the system's security against various forms of attacks. Specific examples of validity checks across different modalities include:

- **Face Recognition:** Implements liveness detection algorithms to differentiate between a real human face and a photograph, video, mask, or other face replicas. Techniques such as analyzing texture patterns, skin reflection, eye blinking, or head movements are used to confirm the presence of a live subject [67, 105, 166, 192].
- **Fingerprint Recognition:** Employs algorithms to detect the presence of a live finger by examining sweat pores, skin elasticity, or pulse, thus preventing spoofing with lifted prints or fake fingers [141, 148, 235, 236].
- **Iris Recognition:** Utilizes pupil dilation response or spontaneous iris texture patterns to ensure the iris image is not a photo or a synthetic image [40, 47, 93, 156, 229].
- **Voice Recognition:** Applies analysis of speech pattern consistency, background noise levels, and temporal voice characteristics to distinguish between live human speech and recorded or synthesized voice attacks [176, 223, 242, 243, 244].

These methods are integral to maintaining the integrity of the biometric authentication process, effectively mitigating risks associated with spoofing and ensuring that only live, genuine biometric data is processed. This layer of security is crucial for the overall reliability and trustworthiness of the decentralized biometric authentication system.

Pre-processing

Following *segmentation/detection*, the pre-processing stage aims to standardize and enhance the quality of biometric data, facilitating more accurate feature extraction. This phase employs various techniques tailored to the specific requirements of each biometric modality:

- **Face Recognition:** Image quality enhancements such as grayscale conversion, contrast adjustment, and histogram equalization are applied to improve facial feature visibility. Additionally, geometric transformations may be used to ensure facial alignment, with eyes positioned on the same level for consistent feature extraction.
- **Fingerprint Recognition:** The focus is on enhancing the fingerprint image to accentuate ridge patterns and minimize noise. Techniques such as ridge enhancement filters and binarization are commonly applied, improving the clarity of minutiae points crucial for matching algorithms.

- **Iris Recognition:** Pre-processing includes iris normalization and unwrapping to transform the circular iris pattern into a rectangular form, facilitating consistent feature analysis. Image enhancement methods are also applied to improve the visibility of iris patterns.
- **Voice Recognition:** Audio signals undergo noise reduction and normalization processes to mitigate background noise and ensure consistent volume levels. Feature extraction techniques, such as Mel-Frequency Cepstral Coefficients (MFCCs), are prepared by segmenting the voice signal into frames for detailed analysis.
- **Gait Recognition:** Video sequences are processed to stabilize the motion and isolate the silhouette of the subject. Normalization techniques adjust for variations in speed and stride, ensuring consistent gait pattern analysis.

This preparatory stage is crucial for ensuring the biometric data is in an optimal state for the feature extraction phase, directly impacting the accuracy and reliability of the biometric authentication system.

Feature extraction

In the feature extraction stage, the essence of biometric data is distilled into a compact, yet informative representation suitable for comparison and authentication. This phase leverages advanced algorithms and techniques to identify and quantify unique biometric characteristics from the pre-processed data. The process varies significantly across different biometric modalities, as illustrated below:

- **Face Recognition:** Utilizes techniques such as geometric feature extraction, where key landmarks on the face (e.g. distances between eyes, nose width, chin shape) are measured, or appearance-based methods, where facial characteristics are captured more holistically, often using deep learning models like Convolutional Neural Networks (CNNs) to generate a facial signature.
- **Fingerprint Recognition:** Focuses on minutiae extraction, identifying unique points on the fingerprint such as ridge bifurcations and endings. Advanced image processing techniques are applied to enhance the fingerprint image, making the minutiae more discernible for accurate template generation.
- **Iris Recognition:** Employs pattern recognition algorithms to analyze the unique patterns in the iris. Features such as trabecular meshwork, rings, furrows, and freckles are encoded into a compact, digital form, often utilizing Gabor filters, or wavelet transforms for texture analysis.
- **Voice Recognition:** Extracts features from voice samples based on the speech signal's frequency, amplitude, and temporal characteristics. Techniques such as Mel-Frequency Cepstral Coefficients (MFCCs) or Deep Neural Networks are used to capture the unique aspects of an individual's voice pattern.

- **Gait Recognition:** Analyzes the sequence of movements or poses constituting an individual's walk. Feature extraction in gait recognition may involve analyzing spatial-temporal patterns or the using motion capture technology to identify unique gait characteristics.

In all modalities, feature extraction aims to transform raw or pre-processed biometric data into a feature set that accurately represents the individual's unique biometric characteristics. These features are then encoded into templates, which are stored or compared against existing templates to verify identity or authenticate users. This stage is critical for the effectiveness and reliability of the biometric authentication process. It requires a careful balance between capturing sufficient detail for accurate identification and maintaining efficiency for real-time processing.

2.1.5 Matcher

The Matcher component plays a crucial role in decentralized biometric authentication systems. Its primary function is to analyze and compare biometric features against stored templates to determine the likelihood of a match. This process involves generating a support value, quantifying the degree to which the presented biometric input aligns with a specific template.

Depending on the system's design and privacy requirements, the Matcher's operations can be localized within the sensor, offloaded to a third party, executed at the service provider's end, or distributed among multiple parties through multi-party computation techniques.

Comparison

The comparison phase assesses the similarity between captured biometric features and stored templates. This process can be classified into two primary models:

- **Closed [155] vs. open set [52]:** In a closed set scenario, all potential users are known and registered within the system. In contrast, an open set context assumes an unrestricted user base, where individuals might not be previously enrolled. The distinction significantly influences the matching algorithm's complexity and the strategies for handling unknown subjects.
- **Supervised vs. unsupervised:** Comparison algorithms may operate under supervised learning, where the system is trained on labeled data, or unsupervised learning, which does not rely on predefined labels. Choosing these approaches affects the system's adaptability and efficiency in recognizing biometric patterns.
- **Template adaptability:** Depending on the operational context, different templates may be employed to enhance matching performance and security. This flexibility allows the system to adjust to varying environmental conditions and threat models.

Pre-select

Before performing an exhaustive search across the entire template database, the Matcher can apply pre-selection criteria to narrow down potential matches. This step enhances system efficiency and reduces computation time through:

- **Identification:** Determining the most likely candidates for a match based on initial biometric input.
- **Soft biometrics [191]:** Utilizing less precise, but still informative, traits such as height or gender to reduce the search space.
- **Non-biometric information:** Employing auxiliary identifiers like passports or ID numbers to further refine the pool of potential matches.

Database

An important component of the Matcher's functionality is its interaction with the biometric template database. This includes mechanisms for:

- **Revocation:** The ability to efficiently remove or update templates within the database, ensuring the system remains current and secure against potential vulnerabilities.

The Matcher ensures accurate, efficient, and secure user authentication by comparing biometric features against stored templates and employing pre-selection techniques.

2.1.6 Aggregator

The Aggregator is tasked with synthesizing support values generated by the Matcher. These values, which may be derived from multiple inputs over time and from different biometrics, cumulatively inform the decision-making process. When the aggregated support exceeds a predefined threshold for a specific action, the system initiates a predetermined action, such as granting or denying access. This component's functionality is crucial for enhancing authentication security, particularly in systems that employ multimodal biometrics, combining traits like fingerprint and palm print through trained neural networks for improved accuracy [109, 184].

Multimodal biometric systems offer a more robust solution than unimodal systems by integrating multiple biometric indicators. For instance, combining fingerprint and iris data [213], or face recognition with RFID tags [15], significantly enhances security and reliability. This integration can occur at various levels, including sensor [118], feature, score, rank, and decision, each offering unique advantages and considerations.

Data quality plays an important role in the aggregation process. High-quality biometric data is essential for reliable authentication, as poor-quality inputs can increase the false acceptance rate (FAR), potentially allowing unauthorized access. To mitigate this risk, the Aggregator can potentially assess the quality of biometric inputs, rejecting or requesting additional data if the quality is below a certain threshold [17, 191]. This assessment is modality-specific, with different criteria for evaluating the quality of fingerprint [31], iris [48], face [142], and other biometric data. Advanced models can also be explicitly trained for quality estimation, further enhancing the system's effectiveness [63].

The Aggregator's approach varies significantly across applications, influenced by factors such as the required security level, operational environment, and user convenience. In some contexts, the system may prioritize one biometric modality over others due to its reliability or ease of use. Alternatively, in more secure settings, the system might require a combination of modalities to authenticate an individual [9]. Continuous authentication presents a dynamic alternative, offering ongoing verification rather than a single, one-time check, thereby enhancing security against spoofing and accommodating intra-class variations and noisy data [161].

In summary, the Aggregator is fundamental to the effectiveness of biometric authentication systems, particularly in multimodal contexts. By intelligently synthesizing support values from various sources and rigorously assessing data quality, the Aggregator ensures a high level of security and reliability. This adaptability to different biometric modalities and operational requirements underscores the complex interplay between technology and application in the realm of biometric authentication.

2.1.7 Performance evaluation metrics and datasets

Evaluating the efficacy of biometric systems involves assessing various metrics that contribute to their overall performance. This subsection focuses on these metrics and their role in ensuring the robustness and reliability of biometric systems in diverse applications.

The primary measure of a biometric system's performance is its accuracy, which is often expressed through the **Equal Error Rate (EER)** [24, 173, 202, 239]. EER indicates the point at which the rates of False Acceptances (FAR) and False Rejections (FRR) converge, offering a balanced metric for system evaluation. Depending on the application's security requirements, emphasis might be placed more on minimizing FAR or FRR to adapt to specific security needs.

Efficiency in biometric systems refers to the speed at which the system processes an identification or verification request and the system's operational demands. This metric is crucial for applications requiring rapid response times without sacrificing accuracy [173]. **Scalability** is another important measure, that assesses a system's capability to handle increasing workloads, such as more users or more sensors, without performance degradation. This is vital in large-scale systems where user bases might expand significantly [174].

Security metrics assess the system's ability to protect user data against unauthorized access and ensure data integrity. The robustness of a biometric system against external attacks and data breaches is a cornerstone of its reliability [173, 174]. **Usability** focuses on the user experience, particularly the ease of interaction with the biometric system and the intuitiveness of its processes. A system with high usability encourages broader acceptance and user compliance [174].

Effectiveness evaluates the real-world applicability of a biometric system, ensuring that it performs well within the expected timeframes under various conditions. This metric is particularly important in scenarios where immediate authentication is necessary [174].

In practical applications, achieving a **balance** among accuracy, efficiency, usability, and security is challenging and often requires trade-offs. The optimal balance is influenced by the specific demands and constraints of the environment in which the system is deployed. For example, the needs of a fast-paced urban subway system differ significantly from those of a high-security border checkpoint. By carefully considering these metrics, developers and users can better understand the strengths and limitations of different biometric systems and select the one that best fits their specific needs.

2.1.8 Security and privacy considerations

In decentralized settings, biometric systems face unique security vulnerabilities and privacy concerns due to their distributed nature. This subsection elaborates on these challenges, underlining the need for rigorous security measures and adherence to ethical guidelines.

Security vulnerabilities in biometric systems

Biometric systems, especially in decentralized architectures, are susceptible to various security threats, each targeting different system components. The main vulnerabilities include:

- **Replay attacks**: Where an attacker reuses previously captured biometric data to gain unauthorized access [193, 194, 195].
- **Disclosure of biometrics**: Unauthorized access leads to sensitive biometric data exposure.
- **Impersonation attacks / spoofing**: Using fake biometric traits to impersonate a legitimate user. Examples include:
  - Liveness attacks on facial recognition systems [191]
  - Fake fingers in fingerprint systems [173]
- **Denial of Service (DoS)**: Overloading the system to prevent legitimate access.
- **Tailgating**: An unauthorized person gaining access by following an authorized person.
- **Replay Attacks**: These occur when an attacker captures and reuses previously recorded biometric data to fraudulently gain unauthorized access to a system [193, 194, 195]. Unlike liveness attacks, replay attacks involve using authentic biometric samples that were recorded during a legitimate interaction, rather than generating fake biometrics.

- **Disclosure of Biometrics**: This refers to the unauthorized access and exposure of sensitive biometric data, which can lead to privacy breaches and further exploitation of the compromised biometric information.
- **Impersonation Attacks / Spoofing**: These attacks involve the use of counterfeit biometric traits to imitate a legitimate user and gain unauthorized access. Examples include:
  - **Liveness Attacks**: Specifically target systems that rely on liveness detection, such as facial recognition, by using techniques like photo, video, or 3D mask attacks to trick the system into accepting a fake biometric as real [191].
  - **Fake Fingers**: In fingerprint systems, attackers may create artificial fingerprints using materials like silicone or gel to impersonate a legitimate user [173].
- **Denial of Service (DoS)**: This involves overwhelming a biometric system with excessive requests or data, thereby preventing legitimate users from accessing the system and causing service disruption.
- **Tailgating**: An attack method where an unauthorized individual gains access to a secured area by closely following an authorized person, exploiting their access without providing valid credentials themselves.

Privacy concerns and ethical implications of biometric data collection

The collection and use of biometric data raise significant ethical and privacy concerns, particularly around consent, data minimization, and the potential for surveillance. In decentralized systems, while data is not centralized, the dispersion can complicate data management, leading to challenges in ensuring all nodes comply with privacy laws and ethical standards. Systems must implement strict data governance to prevent misuse and ensure that users have control over their personal information.

In conclusion, decentralized biometric systems offer benefits over centralized systems, especially in terms of enhanced privacy and reduced risk of mass data breaches, but they require careful consideration of security and privacy to mitigate their unique risks.

Summary

This chapter reviewed the transition from centralized to decentralized biometric authentication systems, discussing both the benefits and challenges of decentralized approaches. While decentralized systems offer potential improvements in privacy, user control, and reduced risk of large-scale data breaches, they also introduce complexities that require advanced security measures, effective data management strategies, and careful coordination. The discussion also touches on the role of multi-modal biometric systems and the integration of various biometric traits. This chapter sets the groundwork for understanding the necessary considerations in developing resilient, user-centric biometric authentication systems.

2.2 Datasets

In this section, we describe the datasets that were used in this PhD thesis. The selection and application of these datasets provide the foundation for evaluating the proposed methodologies and algorithms. Each dataset comes with its unique characteristics, allowing for a comprehensive analysis of face recognition and detection performance under various conditions.

2.2.1 Celebrities in Frontal-Profile (CFP)

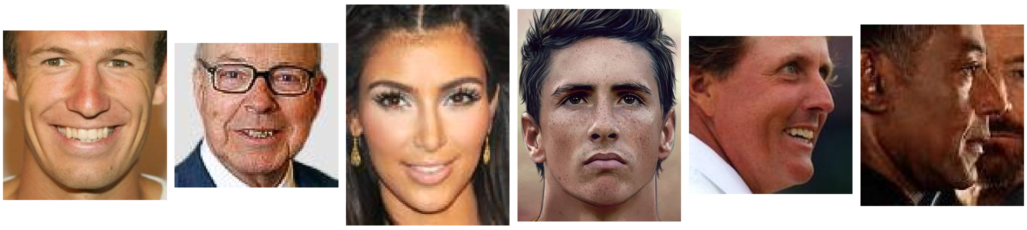


Figure 2.4: Example images of the CFP dataset.

The Celebrities in Frontal-Profile (CFP) [186] dataset consists of images of 500 individuals. Each person has 10 frontal images, making it ideal for studying variations in facial recognition due to changes in expression and pose adjustments. With this dataset we assess the robustness of face recognition algorithms when dealing with occluded face parts (Chapter 3).

2.2.2 Real-world mask dataset

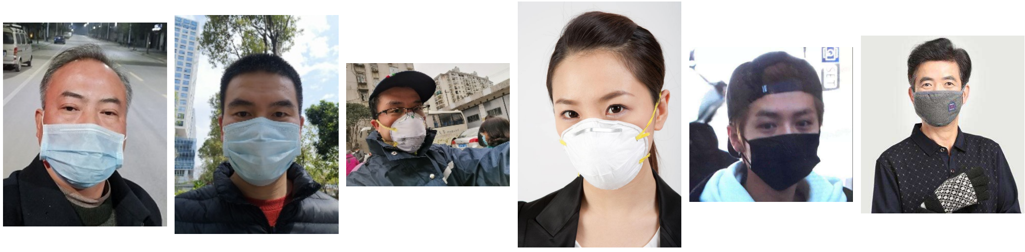


Figure 2.5: Example images of the real-world mask dataset.

This dataset [224] was compiled to specifically address the challenges posed by occluded faces, a scenario increasingly common due to the global pandemic and other factors requiring face coverings. It includes 525 individuals with a total of 2203 images where faces are masked. The variety and realism of these images provided a robust testbed for evaluating the impact of occlusions on face detection and recognition systems (Chapter 3).

2.2.3 WIDER Face

Figure 2.6: Example images of the WIDER face dataset.

The WIDER Face dataset [233] is a comprehensive collection of 32,203 images and 393,703 labeled faces, designed to support research in facial recognition and detection. It encompasses a diverse range of facial variations, including different scales, occlusions, poses, and expressions, ensuring robust and challenging evaluation benchmarks. We utilize this dataset to verify the accuracy of various face detection networks in Section 3.1.1, ensuring a robust and challenging evaluation benchmark.

2.2.4 Labeled Faces in the Wild (LFW)

Figure 2.7: Example images of the LFW dataset.

The Labeled Faces in the Wild (LFW) dataset [94] is a well-known and publicly accessible collection intended for the study and benchmarking of unconstrained face recognition. Consisting of 13,233 images representing 5,749 distinct individuals, this dataset presents a diverse array of conditions relating to illumination, pose, and expression, thereby creating a challenging yet accessible platform for evaluating facial recognition systems. However, a considerable proportion of these images are portrait-like with consistent lighting and favorable angles. This characteristic significantly contributes to the potential for achieving near-perfect accuracy levels with certain advanced face recognition systems, rendering the LFW dataset relatively easy to handle in contrast to other collections with more constrained conditions. Notably, among the entire set, 6,000 image pairs have been specifically arranged to serve as a robust validation subset, further underscoring the LFW's efficacy as a comprehensive tool in the development and evaluation of face recognition methodologies. This dataset, together with CPLFW is used in Section 4 for evaluating the effects of embedding reduction. These two datasets are chosen, because they contain a large amount of real-world, unconstrained images that closely resemble the diverse scenarios encountered in practical applications of face recognition technology.

2.2.5 Cross-Pose Labeled Faces in the Wild (CPLFW)

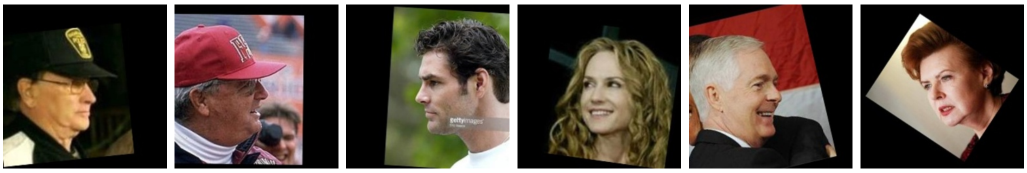


Figure 2.8: Example images of the CPLFW dataset.

The Cross-Pose LFW (CPLFW) dataset [252] provides a more challenging environment for testing face verification technologies due to its focus on pose variations and diverse conditions, including lighting and expressions. Featuring over 11,652 images of 3,000 individuals, CPLFW offers a rich diversity of non-ideal scenarios, representing a stark contrast to the considerable proportion of portrait-like images found in the LFW dataset. This complexity, especially in pose variation, makes achieving high accuracy more challenging for face verification models. CPLFW also includes a validation subset of 6,000 image pairs to facilitate detailed assessments, paralleling the LFW dataset's structure.

Moreover, the CPLFW dataset highlights the variation of poses, which adds another dimension to the challenges faced by face recognition systems. To support comprehensive testing, the CPLFW dataset includes a specially curated validation subset of 6,000 image pairs, mirroring the structure of the LFW dataset.

2.2.6 CelebFaces Attributes (CelebA)

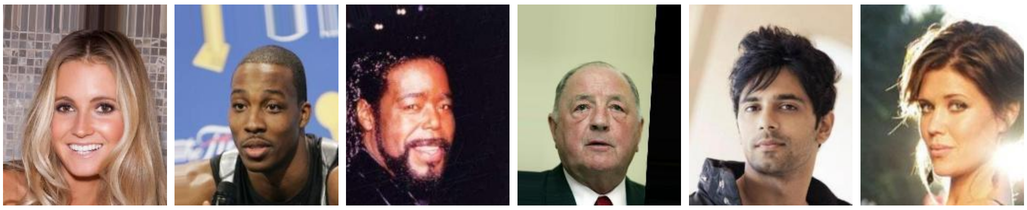


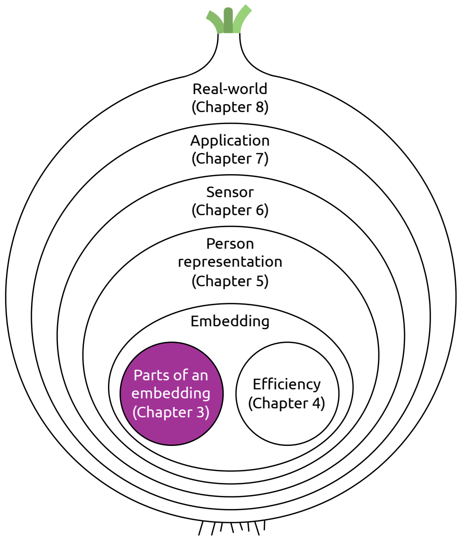
Figure 2.9: Example images of the CelebA dataset.

The CelebFaces Attributes (CelebA) dataset [125] is a large-scale face attributes dataset containing 202,599 celebrity images of 10,177 identities, each with 40 attribute labels. This dataset is particularly valuable for various facial analysis tasks due to its rich diversity in terms of appearance, expressions, and poses, as well as the inclusion of annotated attributes like age, gender, and facial landmarks.

CelebA's extensive annotations enable a wide range of experiments, from face detection and alignment to attribute prediction and landmark localization. This versatility makes it an essential resource for assessing the performance and robustness of face recognition algorithms under different conditions (Chapter 5).

Chapter 3

Understanding facial features in biometric authentication



In the previous chapters, we discussed the significance of biometric systems, explored the potential impact of decentralization, and analyzed various datasets. This foundation sets the stage for our in-depth examination of embeddings—the cornerstone of biometric authentication. Specifically, we will explore how facial features impact face recognition systems. This examination will help us understand what information these embeddings contain and what most affects their performance.

Why focus on facial recognition? Most state-of-the-art biometric systems create an embedding because they avoid re-training their models for every individual to be recognized. They use the embedding of different modalities in a very similar way, even using the same distance metric across modalities. Thus, in this chapter, we will focus on faces as one example of a widely used biometric modality, but we expect other biometric systems to behave similarly. The reason for facial recognition is that it is a widely-used biometric technology due to its non-intrusive nature and broad applicability. Unlike other systems such as fingerprint or iris scanning, it doesn't require direct contact, making it ideal for security, surveillance, and user authentication. Its versatility spans from unlocking smartphones to monitoring public spaces recognizing faces in crowds from a distance. Advances in machine learning and computer vision have enhanced its accuracy and reliability, cementing its role in modern biometrics.

We begin by analyzing the performance of various face-detection and recognition algorithms, as detailed in Section 3.1. This analysis informs our choice of the most suitable face detection and recognition model for this thesis. Next, we evaluate efficiently computable heuristics for enhancing face recognition systems, as discussed in Section 3.2. Face recognition systems are typically trained on high-quality, portrait-like datasets, though recent years have seen more diverse training data. In real-world scenarios, faces are often partially covered or occluded, such as by face masks, sunglasses, or other coverings. This is particularly relevant during virus outbreaks or for medical staff wearing protective gear and raises the question of how these occlusions affect the performance of state-of-the-art face detection algorithms, a topic we address starting in Section 3.3.

Understanding the role of different facial features is essential for making these systems more robust and efficient. By examining the contributions of various facial parts, we aim to identify which features are most critical for accurate recognition. Additionally, this knowledge can improve current algorithms but also enhance privacy by allowing individuals to cover specific parts of their faces to avoid detection by certain algorithms.

3.1 State-of-the-art face pipeline

Foundation

The foundation of this section is the following technical report:

Hofer, Philipp. 2021. Analysis of state-of-the-art off-the-shelve face recognition pipelines. Technical report. Johannes Kepler University Linz, Institute of Networks and Security, Christian Doppler Laboratory for Private Digital Authentication in the Physical World, (March 2021). https://www.digidow.eu/publications/2021-hofer-tr-analysisfacerecognitionpipelines/Hofer_2021_AnalysisFaceRecognitionPipelines.pdf

To advance our research into the behavior of face recognition models, we must first determine which models to use. Face recognition pipelines are continually evolving, with numerous new publications each year [106, 136, 154, 188, 224]. This section aims to provide an overview of a modern pipeline and recommend a state-of-the-art approach that balances accuracy and performance, even on low-end hardware such as the Jetson Nano[1], or a Raspberry Pi[2] with a Coral USB Accelerator[3].

State-of-the-art face recognition pipelines typically involve two primary tasks (c.f. Fig. 3.1):

- **Face detection:** Identifying and locating faces within images.
- **Face recognition:** Mapping detected faces to specific individuals.

[1] https://developer.nvidia.com/embedded/jetson-nano
[2] https://www.raspberrypi.com
[3] https://coral.ai/products/accelerator

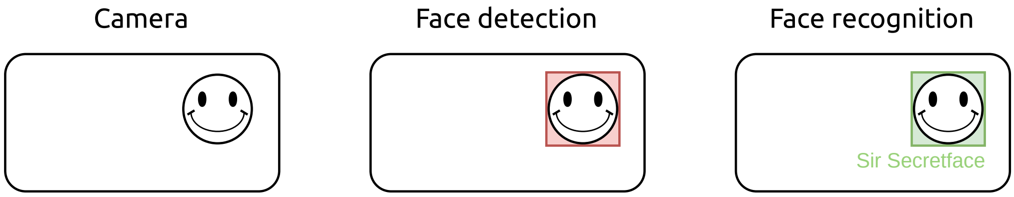


Figure 3.1: Difference between face detection and face recognition.

To develop an effective real-time system, the pipeline must process a sufficient number of frames per second (FPS) to ensure smooth and responsive performance. Stewart et al. [199] indicate that real-time systems typically need to process at least 3–5 frames per second to be considered responsive and effective in dynamic environments. This performance benchmark is crucial for maintaining a smooth and responsive user experience. Notably, while the cited study is from 2001, its findings continue to be relevant for establishing baseline requirements in real-time system performance.

3.1.1 Face detection

We analyze three popular state-of-the-art face detection models: Retinaface [53], MTCNN [241], and Faceboxes [245]. We aim to assess which model delivers the most accurate and efficient performance for practical applications. To facilitate a fair and consistent comparison, we utilized the WIDER Easy Face dataset [233], chosen for its diversity and representativeness of real-world scenarios. For a comprehensive description of this dataset, please refer to Section 2.2.3.

The accuracy of the models was evaluated with the following results:

- Retinaface: 94.2 %
- MTCNN: 91.0 %
- Faceboxes: 86.3 %

Subsequently, we assessed the processing speed of each model by timing their performance on a 1080p image using an *Intel Core i5-8265U CPU*:

- Retinaface: 750 ms (1.3 FPS)
- MTCNN: 550 ms (1.8 FPS)
- Faceboxes: 35 ms (28 FPS)

This evaluation reveals a notable trade-off between speed and accuracy among the models. FaceBoxes demonstrates superior speed, making it highly efficient for scenarios where processing speed is critical. However, this speed comes with a limitation: FaceBoxes performs optimally only on high-quality images where the face is large and directly facing the camera.

In contrast, both Retinaface and MTCNN not only provide higher accuracy but also compute five facial landmarks necessary for advanced face recognition tasks.

Looking ahead, there are several strategies to enhance model speed without significantly sacrificing accuracy, which we explore in greater detail in Chapter 6.

- **Adopt a smaller backbone network:** Face detection networks use *backbone networks* for extracting features from images. The size of the backbone network, primarily its depth, significantly affects inference time. Using a leaner network can reduce processing time, though it might reduce accuracy.
- **Reduce image resolution:** Lowering the resolution of images can speed up processing times, although this often results in lower image quality, potentially undermining the utility of high-resolution cameras.
- **Selective detection**: Running the face detection algorithm only on specific parts of an image, such as areas with movement or previously detected faces, can improve efficiency.
- **Simplified initial detection:** Using a basic model to generate initial face proposals and then applying the comprehensive model only on these cropped regions can optimize both speed and resource use.

3.1.2 Face recognition

Comparing a face against numerous others efficiently without re-training the network requires extracting a numerical representation, known as an embedding. Recent publications, such as Arcface [52], SphereFace [123], and CosFace [221], commonly use an embedding size of 512 elements with 32-bit floating points (in total 16 kB).

We focus on Arcface and FaceNet over other algorithms due to several factors. Firstly, Arcface has established itself as a state-of-the-art method for face recognition, known for its high accuracy and robust performance across diverse datasets. FaceNet, on the other hand, is widely recognized for its efficiency and versatility, particularly in embedding generation and real-time applications. The combination of these two methods allows us to explore both cutting-edge accuracy and practical deployment efficiency.

Therefore, we conducted a detailed performance comparison of Arcface and FaceNet across various hardware platforms, focusing on the speed and efficiency with which they process face embeddings. The results of this evaluation, presented in Table 3.1, highlight the strengths of each algorithm in different contexts, providing valuable insights into their practical applications.

Enhancing performance using a GPU, whether on a laptop or an embedded device such as a Jetson Nano, would improve these speed metrics. However, since our research focuses on assessing these models' relative efficiency under similar conditions, we did not incorporate GPU enhancements into our testing protocol. This approach remains aligned with our primary objective of evaluating model performance in a standard computational environment, which is crucial for applications in less resource-intensive settings.

Table 3.1: Speed comparison of 2 state-of-the-art face-recognition algorithms.

	Arcface	FaceNet
Laptop[4]	0.21 s/ face embedding	0.17 s/ face embedding
Pi 3	3.5 s/ face embedding	3.1 s/ face embedding
Pi 4	2 s/ face embedding	1.5 s/ face embedding

To measure accuracy, we relied on the findings of Firmansyah et al. [65], who evaluated these algorithms on the Labeled Faces in the Wild (LFW) dataset:

- FaceNet: 99.20 %
- Arcface: 99.41 %

3.1.3 Summary

In this section, we have explored the typical components and performance of face recognition systems, focusing on both face detection and -recognition.

Face detection: The findings reveal a trade-off between accuracy and speed for three state-of-the-art face detection models (Retinaface, MTCNN, and Faceboxes).

- Retinaface: Offers the highest accuracy (94.2 %) but operates at the slowest speed (1.3 FPS).
- MTCNN: Provides good accuracy (91.0 %) with moderate speed (1.8 FPS).
- Faceboxes: Delivers the fastest speed (28 FPS) but with the lowest accuracy (86.3 %).

The trade-offs highlight the need for tailored solutions based on application requirements. Enhancements such as using smaller models and targeted detection areas can further optimize performance.

Face recognition: We conducted a comparison between two face recognition algorithms, Arcface and FaceNet, focusing on their speed and accuracy. Given that their accuracy is nearly identical, our analysis emphasized processing speed:

- Arcface: Processes face embeddings in 0.21 seconds on a laptop, 3.5 seconds on a Pi 3, and 2 seconds on a Pi 4.
- FaceNet: Processes face embeddings in 0.17 seconds on a laptop, 3.1 seconds on a Pi 3, and 1.5 seconds on a Pi 4.

These comparisons demonstrate the relative efficiency and effectiveness of the models, underscoring their suitability for real-time applications even without GPU acceleration.

Decision

Given the high accuracy and reasonable performance balance, Retinaface and Arcface emerge as the recommended default models for robust face detection and -recognition tasks and will be used in this thesis.

3.2 Heuristics for successful face pipeline

Foundation

The foundation of this section is the following technical report:

Hofer, Philipp. 2021. Face recognition: Increase accuracy by filtering images with heuristics. Technical report. Johannes Kepler University Linz, Institute of Networks and Security, Christian Doppler Laboratory for Private Digital Authentication in the Physical World, (July 2021). https://www.digidow.eu/publications/2021-hofer-tr-increasefacerecognitionaccuracy/Hofer_2021_IncreaseFaceRecognitionAccuracy.pdf

This section explores the development and validation of simple heuristics aimed at efficiently distinguishing between successful and unsuccessful face recognition attempts. Our objective is to enhance the preprocessing steps within face recognition pipelines by incorporating these heuristics, which are designed to be both effective and computationally economical. This approach not only simplifies the preprocessing phase but also reduces the computational resources required, leading to cost-effective and faster processing.

Expanding upon the analysis of various face detection and recognition algorithms presented in Section 3.1, this discussion shifts towards heuristic methods that could potentially elevate the performance of face recognition systems. Ideally, these heuristics also enrich our understanding of the role that distinct facial features play in successfully detecting faces.

We consider the following heuristics:

1. **Eye distance relative to face width:** The distance between the eyes, when scaled by the face width, indicates the face's alignment with the camera. Full-frontal faces have a more considerable distance between the eyes relative to the face width, whereas profiles show a smaller distance. This metric helps determine whether the face is fully visible or partially obscured.

2. **Eye-mouth distance relative to face height:** The vertical distance from the center of the eyes to the mouth, scaled by the face height, provides information about the face's tilt and angle. This heuristic is important for assessing the face's orientation, which affects recognition accuracy.

3. **Face area:** The overall size of the face in pixels is thought to correlate with the amount of detailed information available for recognition. More prominent faces are expected to yield better recognition accuracy due to the richer detail they contain.

3.2.1 Experimental setup

To evaluate these heuristics, we conducted an experiment using a dataset collected with a camera in a controlled environment. The dataset includes images of 13 distinct individuals, each with between 5 and 210 images, averaging 79.3 images per person. Each individual's template image was manually taken, and the embeddings were calculated:

```python
def get_template_embs(path):
    templates = dict()
    for person in os.listdir(template_path):
        person_path = os.path.join(template_path, person)
        person_name = os.path.splitext(person)[0]
        templates[person_name] = rec.get_template_embedding(person_path)
    return templates
templates = get_template_embs(template_path)
```

The recognition process sorts the results into correct and incorrect identifications using a specific threshold value set at 1.2. This threshold was chosen based on empirical analysis, as it effectively balances the number of false positives (incorrectly identifying an incorrect match as correct) and false negatives (failing to identify a correct match) across various datasets. It is recommended that this threshold be fine-tuned for real-world applications or deployments to optimize performance for specific conditions or requirements. More details on adjusting and optimizing the threshold can be found in Chapter 7. With our threshold of 1.2, we achieve an initial accuracy rate of 25.3 %, with 261 correct identifications and 770 incorrect ones.

```python
correct = []
wrong = []
for person_folder in person_folders():
    template = templates[os.path.basename(person_folder)]
    for img in os.listdir(person_folder):
        img_path = os.path.join(person_folder, img)
        emb = rec.get_emb(img_path)[0]
        if rec.get_score(template, emb) < threshold:
            correct.append(img_path)
        else:
            wrong.append(img_path)
```

3.2.2 Detailed analysis

- **Eye distance:** We analyzed the distance between the eyes, scaled by the face width, across successful and unsuccessful recognition attempts. A threshold of 8.5 % of the face width was identified, below which recognition failed (Fig. 3.2). This suggests that ensuring a minimum eye distance is important for successful recognition. Discarding faces where the eye distance is less than e.g. 8 % of the total face width could reduce false positives without adversely affecting performance.

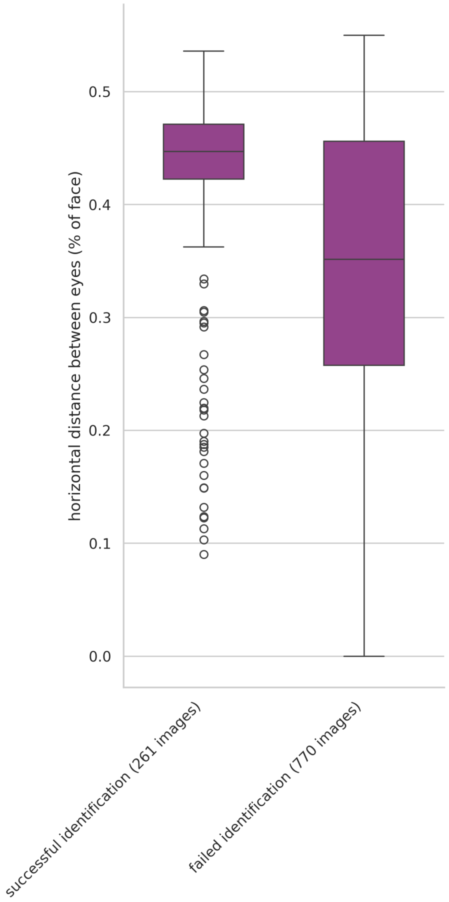


Figure 3.2: Horizontal distance between eyes scaled by face width, grouped by successful identification.

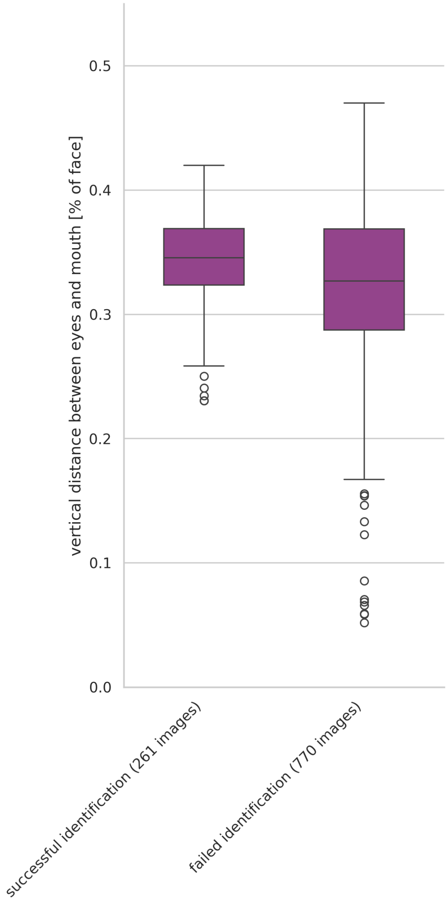


Figure 3.3: Vertical distance from the center of the eyes to the mouth scaled by face height and grouped by successful identification.

- **Eye-mouth distance:** Similarly, the distance from the center of the eyes to the mouth, scaled by the face height, was examined. A threshold of 20 % of the face height was established, below which recognition often failed (Fig. 3.3). This heuristic helps filter images where the face angle might lead to recognition errors.
- **Face area:** To test the hypothesis that more prominent faces contain more information and thus yield better recognition accuracy, we analyzed each image's face area in pixels. The results were grouped by successful and unsuccessful recognition attempts. Initially, no clear threshold was identified (Fig 3.4). No clear threshold can be identified even after removing outliers (Fig. 3.5). However, it was noted that more prominent faces generally provided better recognition results. Further studies could focus on artificially reducing image sizes to establish a lower bound for effective recognition.

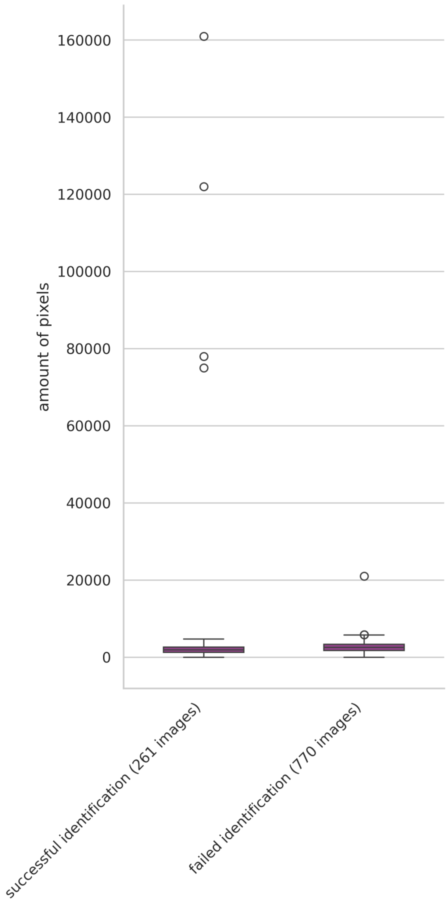


Figure 3.4: Face area (in pixels) analyzed for recognition accuracy, showing no clear threshold for success.

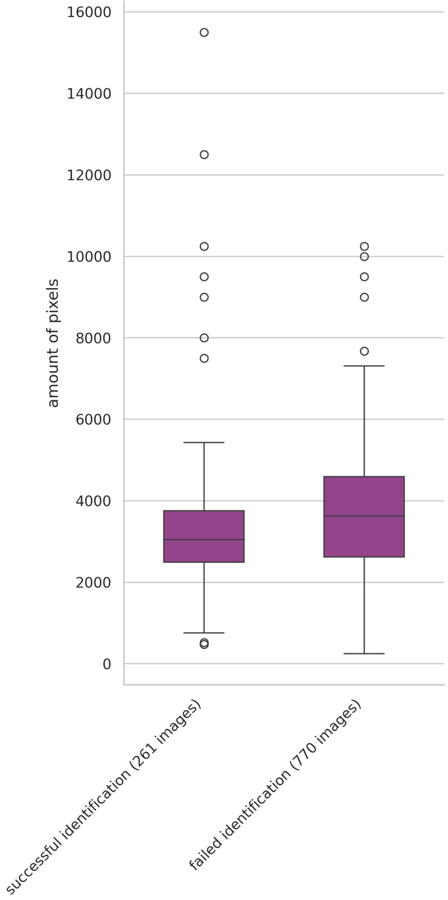


Figure 3.5: Face area (in pixels) analyzed for recognition accuracy with outliers removed, still showing no clear threshold for success.

Summary

Incorporating these heuristics, as detailed in our prototype implementation described in Chapter 8, into face recognition pipelines can preemptively filter out poor-quality images, enhancing overall system accuracy and efficiency.

3.3 Dataset adaptation for key facial feature analysis

Foundation

The foundation of the subsequent sections is the following paper:
Hofer, Philipp, Michael Roland, Philipp Schwarz, Martin Schwaighofer, and René Mayrhofer. 2021. Importance of different facial parts for face detection networks. In *2021 9th IEEE International Workshop on Biometrics and Forensics (IWBF)*. IEEE, Rome, Italy, (May 2021), pp. 1–6. DOI: 10.1109/IWBF50991.2021.9465087

After analyzing which face pipeline to use and examining simple heuristics, we now turn our focus back to occlusion. In this chapter, we differentiate between three distinct tasks related to face recognition. Face detection and face recognition have already been introduced in Section 3.1. In addition, we now consider:

3. *Face mask detection:* These networks are similar to *face detection algorithms*, with the major difference that instead of a single class, two classes are detected: faces wearing masks and faces not wearing them. The output are bounding boxes for both classes.

The training dataset plays an important role in current face detection and -recognition tools, and largely influences their accuracy. Face recognition tools are trained with millions of face images (popular implementations of Arcface [52] use 5.8 million images, FaceNet [181] 200 million images). The datasets used to train current state-of-the-art face recognition tools do not mention the use of images of people with face masks, and thus, we suspect that only a small fraction of the training images contain a person wearing a face mask. To support this claim, we ran an off-the-shelve face mask detection algorithm [91] (with a threshold value of 0.5) on the VGGFace2 dataset [27], a large dataset (3,141,890 images), which is commonly used to train face detection and -recognition networks. The face mask detection algorithm classified only 12,671 images as *person with facemask*. Through manual verification of these proposals, we found 12,616 false positives. Only 34 showed people wearing medical face masks and 21 showed people with mouths and noses covered with fabrics. Thus, only 0.018 % (55/3,141,890) of all images in the dataset [27] depict people with face masks.

Because of this discrepancy of models not seeing masked faces while training and people wearing face masks in real-world settings, this chapter analyzes the performance of three off-the-shelf face detection algorithms (MTCNN [241], Retinaface [50], and DLIB [107]) in this setting. MTCNN uses a three-stage pipeline to exploit the inherent correlations between face detection and face alignment using deep convolutional neural networks [250]. In contrast to MTCNN's multi-stage pipeline, Retinaface is a robust single-stage face detector that employs only lightweight backbone networks while still achieving state-of-the-art accuracy. To analyze a completely different architecture, we included DLIB's face detection algorithm, which uses histogram of oriented gradients (HOG) to detect faces. ,These networks have been trained with a negligible amount—if any—of faces with masks and are evaluated against images where parts of the face are occluded, e.g., by putting on a face mask. The main research question is how different facial parts influence the accuracy of state-of-the-art face detection networks.

Source code

> The code for modifying the dataset and evaluating is available at https://github.com/mobilesec/occluded-facedetection-performance and in Chapter A.

3.3.1 Related work

As Damer et al. [42] stated, the *detection of occluded faces is a well-studied issue in the computer vision domain.* Especially since the Covid-19 outbreak, a lot of research results have appeared in this area.

In order to increase face detection performance on occluded faces, Zhang et al. [248] propose a hard image mining strategy. This results in more emphasis on hard samples, which models reality more closely. Furthermore, to detect partially occluded faces, Zeng et al. [60] introduced the *triplet loss* training strategy.

In order to objectively analyze current performance of face detection algorithms on masked faces and to increase the accuracy for future face recognition algorithms, new datasets with masked faces have been proposed [224]. However, these datasets are still in the early stages as they feature only a 4-digit number of faces. Thus, they are between 3 (w.r.t. MS1M [72]) and 4 (w.r.t. FaceNet) orders of magnitude smaller than current face detection datasets without masks.

Due to the current increase in popularity of face masks, literature tries to improve performance despite having large parts of the face covered [197]. This is an ongoing research activity. Many current popular face detection algorithms are not yet specifically trained on occluded faces. Therefore, this chapter studies the performance of these popular face detection algorithms.

Every face recognition algorithm depends on a face detection algorithm [210]. Thus, if the face detection algorithm does not detect a face, any face recognition algorithm is rendered useless. A study for face recognition algorithms has been performed by NIST[5], where they published an evaluation of the performance of current state-of-the-art face recognition algorithms [145], without being fine-tuned for masks. This is a reasonable assumption since it holds for most of

[5] https://www.nist.gov/

the currently used face recognition systems. NIST also plans to perform a similar experiment with algorithms specifically tuned to recognize people wearing masks [145]. Similarly, our experiments evaluate the performance of popular face detection algorithms, consequently focusing on the preprocessing stage that images need to pass in order to even be considered for later face recognition.

3.3.2 Experimental results

Our goal is to evaluate the effects of occlusions on face detection and recognition performance by adapting the dataset to highlight key facial features. This involves systematically modifying images to obscure specific facial regions and analyzing the impact on detection accuracy. For this, we use two different datasets: CFP and real-world mask dataset. For a description of these refer to Sections 2.2.1 and 2.2.2, respectively.

In order to check which part of the face is most important for face detection (further discussed in Section 3.4), we modified the images of the first dataset [186] by excluding certain areas. There are two main strategies employed:

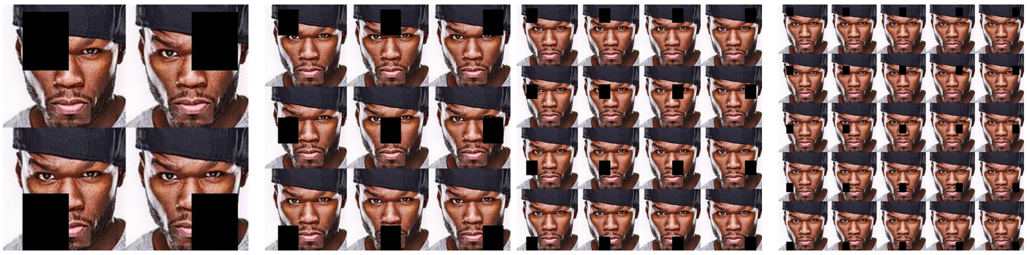
1. Overlaying a grid in various sizes over the face and blacking out one cell at a time. To be able to clearly see what modification has taken place, the resulting modifications for one randomly selected person are visualized in Fig. 3.6.
2. Removing facial landmarks:
   - eye(s) (Fig. 3.7a, 3.7c, and 3.7e),
   - nose (Fig. 3.7i), or
   - mouth (Fig. 3.7g).

   In order to be able to objectively measure the impact of these landmarks on the face detection accuracy, for each of these settings, we create another modification where the same amount of area is blacked out on a random other part of the face (modifications ending with *-not*).

   Since the sizes of the images and therefore the faces vary significantly, the size of the blacked-out area is proportional to the size of the face. The specific proportions have been empirically chosen such that the landmark is sufficiently removed.

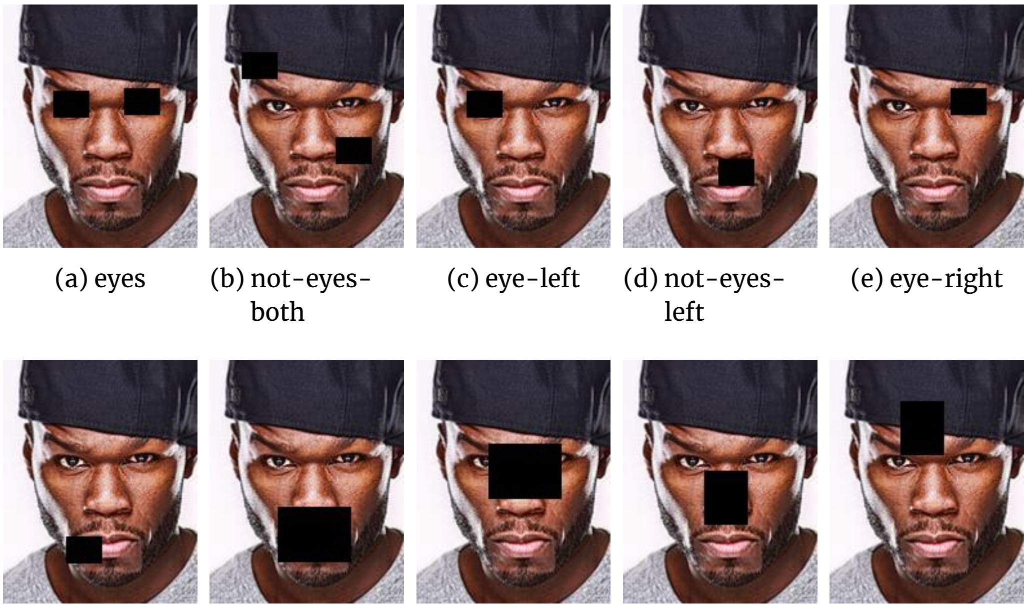
Manually modifying 5,000 images for all these settings is not feasible. Therefore, we automate the creation of these modifications, ensuring consistency and efficiency in the dataset adaptation process. There are two requirements for creating these modifications:

1. *Background vs. face:* Even though all people are displayed in a portrait style where only one person is visible and takes up the majority of the space, the exact location of the face is not constant. For creating the modified versions *grid* and *grid-mask*, we do not want to distort the results by blackening out background pixels instead of pixels belonging to the face.



(a) 2x2 grid (b) 3x3 grid (c) 4x4 grid (d) 5x5 grid

Figure 3.6: Proposed modifications of the CFP dataset concerning blacking out grid cells in various sizes.



(a) eyes (b) not-eyes-both (c) eye-left (d) not-eyes-left (e) eye-right

(f) not-eyes-right (g) mouth (h) not-mouth (i) nose (j) not-nose

Figure 3.7: Proposed modifications (landmarks-*) of the CFP dataset concerning blacking out landmarks of the face.

2. *Face landmarks:* In order to be able to remove face landmarks, we need to know their location.

MTCNN returns the keypoints for the landmarks, and is, therefore, utilized in this chapter to automatically modify the datasets. The ground truth for all experiments in this subsection are 4.978 faces, excluding 22 faces for which MTCNN could not detect a face on the unmodified dataset. We defined the size of the rectangle through empirical experiments to cover the respective landmarks properly. For example, for removing the eyes, we chose the rectangles' width to be 25 % and the height to be 15 % of the face width, as these values seem to adequately cover the eyes in most instances. Even though we do not expect the specific values used in this chapter to impact the results significantly, they are easily retrievable for every setting through the provided GitHub repository link.

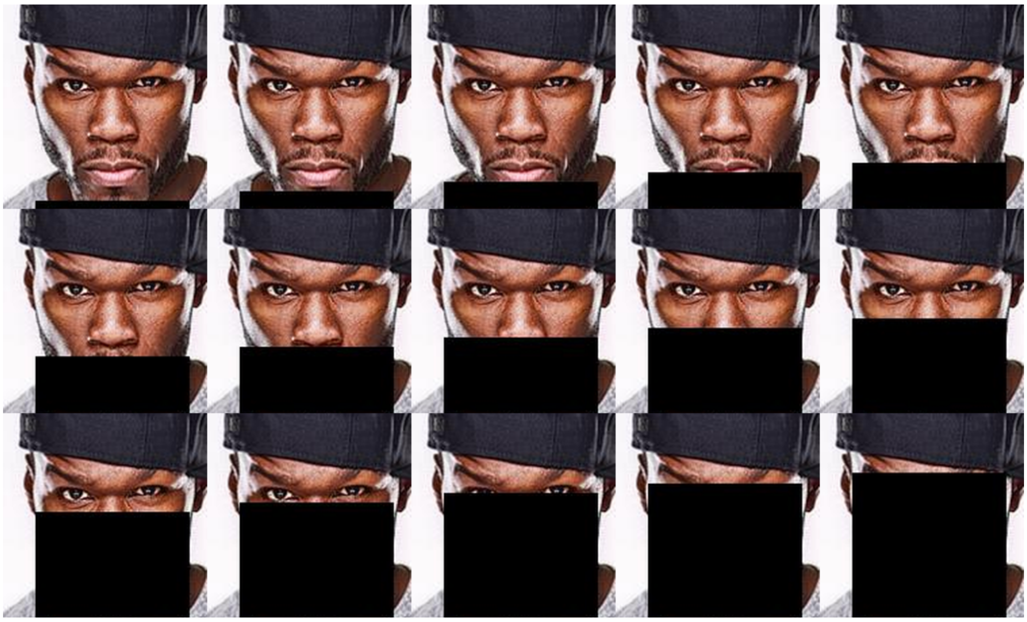


Figure 3.8: Proposed modifications (grid-mask-{00-15}) of the CFP dataset concerning simulating a face mask.

3.4 Experimental results

In order to verify which face regions are most important for face detection, we check the accuracy of three state-of-the-art face detection algorithms on computer-modified images and real-world images of people wearing face masks.

3.4.1 Computer modified images from the CFP dataset

We feed our dataset of modified images from the CFP dataset into MTCNN, Retinaface, and DLIB and analyze their accuracy.

Baseline

To be able to compare the performance of the face detection algorithms on differently modified CFP datasets, we first calculate the accuracy of the three analyzed algorithms on the dataset without any modification. In this chapter, we are interested in correctly detecting the face. We are not differentiating between false positives (i.e. wrongly classifying part of the image as a person) and false negatives (i.e. not detecting a person). They both count as *misclassification* and thus reduce the accuracy equally. From the 5,000 images, between 99.2 % (DLIB, 4960 / 5000 images) and 99.74 % (Retinaface, 4987 / 5000 images) of all visible humans are successfully detected.

Table 3.2: Misclassification rates for grid-2 modification.

Area	Misclassification rate		
	MTCNN	Retinaface	DLIB
Top left corner (00)	14.1 %	9.96 %	20.81 %
Top right corner (01)	18 %	9.94 %	28.65 %
Bottom left corner (02)	3.6 %	4.3 %	39.94 %
Bottom right corner (03)	9.78 %	6.29 %	63.1 %

Table 3.3: Accuracy for the flipped image in the *grid*-2 setting.

	MTCNN			Retinaface			DLIB		
	Correct	0 faces	2 f.	Cor.	0 f.	2 f.	Cor.	0 f.	2 f.
grid-flip-2/00	4303	674	2	4479	495	5	3951	1024	4
grid-flip-2/01	4139	836	4	4465	508	6	3616	1360	3
grid-flip-2/02	4758	218	3	4736	230	13	2899	2076	4
grid-flip-2/03	4571	406	2	4659	314	6	1727	3245	7

3.4.2 Grid

The modifications are named after the amount of both horizontal and vertical cells.

Grid-2 In this setting, we blacked-out a quarter of the face. There is an interesting difference in accuracy between these quarters, as shown in Table 3.2. These results suggest that the top half of the face is more important for face detection, as they have a higher misclassification rate. Interestingly, in all three face detection algorithms, the bottom left corner has a significantly lower misclassification rate than the other 3 corners. This could be due to two facts:

1. The modified CFP dataset is biased, and the bottom left quarter is not as informative as the remaining ones. Therefore, the face detection algorithms (correctly) do not emphasis this part of the image.
2. The pre-trained face detection models are biased, e.g., using a biased training dataset.

In order to exclude the first possible explanation, we created another modification of the dataset by flipping the image vertically (*grid-flip*-2). If the first statement is true, we also expect the misclassification rate to flip. Table 3.3 shows that this is not the case. The misclassification rate is still lowest if the bottom left quarter is blacked out.

Grid-3 For all variations except for blacking out the middle cell, all three algorithms perform pretty well:

Table 3.4: Accuracy for the flipped image in the *grid-4* setting.

	MTCNN			Retinaface			DLIB		
	Correct	0 faces	2 f.	Cor.	0 f.	2 f.	Cor.	0 f.	2 f.
grid-flip-4/00	4971	6	2	4961	14	4	4930	46	3
grid-flip-4/01	4925	52	2	4912	64	3	4861	115	3
grid-flip-4/02	4925	50	4	4914	60	5	4837	139	3
grid-flip-4/03	4965	11	3	4951	24	4	4924	52	3
grid-flip-4/04	4942	35	2	4959	16	4	4801	175	3
grid-flip-4/05	4708	259	12	4901	71	7	4659	317	3
grid-flip-4/06	4378	597	4	4925	50	4	4418	557	4
grid-flip-4/07	4916	61	2	4953	24	2	4813	163	3
grid-flip-4/08	4955	20	4	4963	13	3	4840	137	2
grid-flip-4/09	4861	116	2	4922	48	9	4768	208	3
grid-flip-4/10	4814	159	6	4921	51	7	4722	253	4
grid-flip-4/11	4946	30	3	4953	21	5	4816	158	5
grid-flip-4/12	4968	6	5	4966	11	2	4895	80	4
grid-flip-4/13	4954	23	2	4942	33	4	4857	120	2
grid-flip-4/14	4951	25	3	4948	28	3	4862	113	4
grid-flip-4/15	4969	7	3	4960	14	5	4872	103	4

1. MTCNN: 0.3–5.26 % misclassification rate
2. Retinaface: 0.54–2.53 % misclassification rate
3. DLIB: 1.19–16.65 % misclassification rate

Interestingly, the last case with a black middle cell achieves a significantly larger misclassification rate: 41.4 % (MTCNN), 12.3 % (Retinaface), and 55.0 % (DLIB). This might indicate that the nose plays an important role in face detection, which we will test in Section 3.4.2 in more detail.

Grid-4 and Grid-5 The analysis reveals a consistent pattern of increased misclassification rates for cells that encompass or intersect with the nose:

- As illustrated in Table 3.4, which presents the results for a *4x4* grid configuration, there is a notable decline in accuracy for the four settings where the occluded area overlaps with the nasal region: */5*, */6*, */9*, and */10*.
- Similarly, in the *5x5* grid configuration, as shown in Table 3.5, the settings */11*, */12*, */13*, */16*, */17*, and */18* exhibit a drop in accuracy due to the occlusion of the nasal region.

Table 3.5: Accuracy for the flipped image in the *grid-5* setting.

	MTCNN			Retinaface			DLIB		
	Correct	0 faces	2 f.	Cor.	0 f.	2 f.	Cor.	0 f.	2 f.
grid-flip-5/00	4973	4	2	4968	7	4	4934	42	3
grid-flip-5/01	4925	25	2	4954	23	2	4922	54	3
grid-flip-5/02	4920	56	3	4949	26	4	4846	129	4
grid-flip-5/03	4958	18	3	4949	26	4	4898	78	3
grid-flip-5/04	4973	4	2	4967	8	4	4934	42	3
grid-flip-5/05	4964	12	3	4966	9	4	4887	89	3
grid-flip-5/06	4932	44	3	4960	16	3	4906	69	4
grid-flip-5/07	4618	354	7	4954	19	6	4704	271	4
grid-flip-5/08	4826	149	4	4960	15	4	4863	112	4
grid-flip-5/09	4966	9	4	4963	13	3	4900	76	3
grid-flip-5/10	4957	20	2	4968	9	2	4868	109	2
grid-flip-5/11	4882	94	3	4924	45	10	4756	219	4
grid-flip-5/12	4786	190	3	4940	29	10	4745	231	3
grid-flip-5/13	4751	224	4	4927	47	5	4722	254	3
grid-flip-5/14	4936	40	3	4960	14	5	4852	124	3
grid-flip-5/15	4970	5	4	4968	8	3	4891	86	2
grid-flip-5/16	4943	34	2	4964	12	3	4859	118	2
grid-flip-5/17	4907	69	3	4947	24	8	4878	99	2
grid-flip-5/18	4916	61	2	4959	15	5	4877	100	2
grid-flip-5/19	4965	9	5	4965	9	5	4873	103	3
grid-flip-5/20	4974	3	2	4968	7	4	4919	57	3
grid-flip-5/21	4970	7	2	4958	18	3	4874	103	2
grid-flip-5/22	4964	13	2	4953	23	3	4891	86	2
grid-flip-5/23	4976	9	3	4960	16	3	4890	87	2
grid-flip-5/24	4976	2	1	4970	5	4	4897	79	3

Table 3.6: Results of three face detection algorithms (MTCNN, Retinaface, and DLIB) on real-world mask dataset [224].

	MTCNN			**Retinaface**		**DLIB**	
	0 faces	2 f.	3 f.	0 f.	2 f.	0 f.	2 f.
RMFD [224]							
(2203 images)	1196	1	1	1250	1	2129	1

Area around landmarks

Eye region The eye region is critical for face recognition [175]. Thus, it might also be of particular importance for face detection. Therefore, as introduced in Section 3.3 we modified the CFP dataset, such that features around the eye region are removed.

MTCNN, Retinaface, and DLIB achieve approximately the same accuracy (97.2 %, 99.4 %, and 98.4 %, respectively) if the area around both eyes are removed.

Suppose the eye region plays a more important role than other parts of the face. In that case, the amount of errors (false positives and false negatives) of face detection algorithms will be higher if compared to a dataset where rectangles with the same size are inserted on random positions (*eyes-both-not*). Our experiments contradict this argument, as all three algorithms detect between 1.2 (Retinaface) and 7.3 percent points (DLIB) *more* faces if the rectangles are randomly located. This suggests that other parts of the face are more important for face detection accuracy. One possible explanation is that people in the *eyes-both* dataset look like they are wearing sunglasses, which face detection algorithms have already seen in the training phase.

Similar results are obtained if we occlude a single eye (datasets *eyes-{left|right}[-not]*).

Mouth If we remove the mouth, we see similar results as when removing the eye region. Compared to the version where the *mouth* is covered, the face detection algorithms detect between 4.6 (Retinaface) and 14.1 (DLIB) percent points *more* faces if the rectangles are randomly distributed (*mouth-not*). Therefore, the mouth does not seem to have a higher importance in face detection algorithms.

Nose If we evaluate the face detection algorithms on images where the nose has been blacked out, the algorithms achieve an accuracy of only 72.4 % (MTCNN), 94.3 % (Retinaface), and 58.2 % (DLIB). If we remove a rectangle of similar size, accuracy increases to 98.3 %, 98.5 %, and 94.6 %, respectively. One (partial) reason for this significant difference (especially considering MTCNN and DLIB) might be that the nose is in the very center of the face.

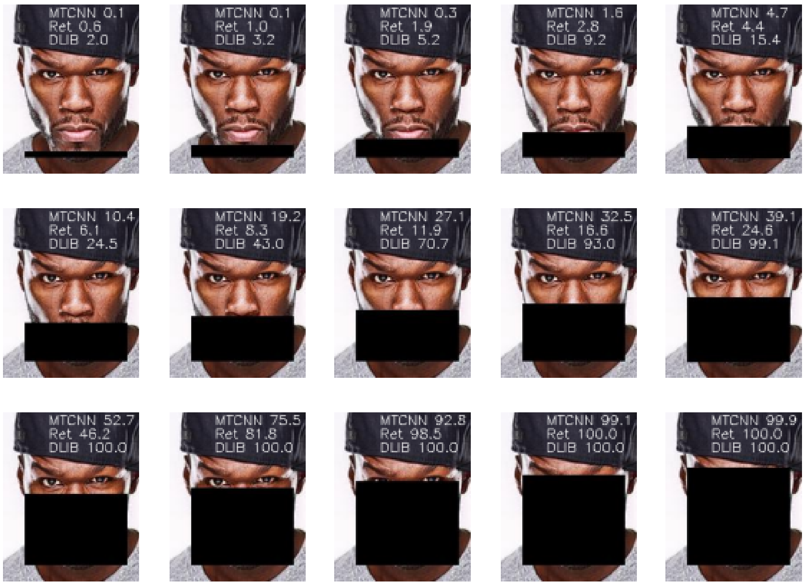


Figure 3.9: Misclassification results in percentage for simulated face mask modification.

In general, with an average accuracy of 97.4 % Retinaface seems to handle occluded faces significantly better than MTCNN (91.6 %) and DLIB (87.7 %).

3.4.3 Mask

In this modification we simulated a face mask of various sizes. As expected, there is a positive correlation between the size of the face mask and the misclassification. The results are shown in Fig. 3.9.

3.4.4 Real world mask images

So far, we have only considered face occlusions which a computer has generated. In this subsection, we evaluate the performance on real-world mask images from RMFD [224]. A detailed description of the dataset is found in Section 2.2.2. MTCNN and Retinaface both detected around 45 % of the faces; DLIB only detected 3 % of all faces (Table 3.6). One possible reason for these low accuracy rates is the challenging dataset. Some people wear both a face mask and sunglasses, resulting in most of the face being occluded.

3.5 MTCNN face-in-face malfunction

Since many state-of-the-art face recognition tools, such as Arcface and SphereFace, recommend using MTCNN, we evaluated its performance on the real-world masked dataset [224]. As shown in Table 3.6, face detection worked for 46 % (1007/2203) of the images from the real-world mask dataset

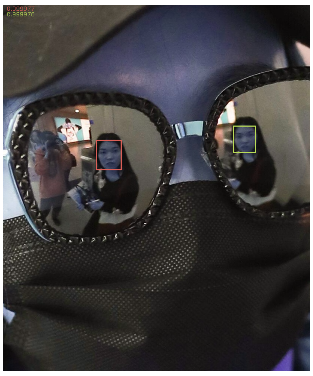


Figure 3.10: MTCNN detects the reflected person in both lenses while missing the person wearing the eyeglasses.

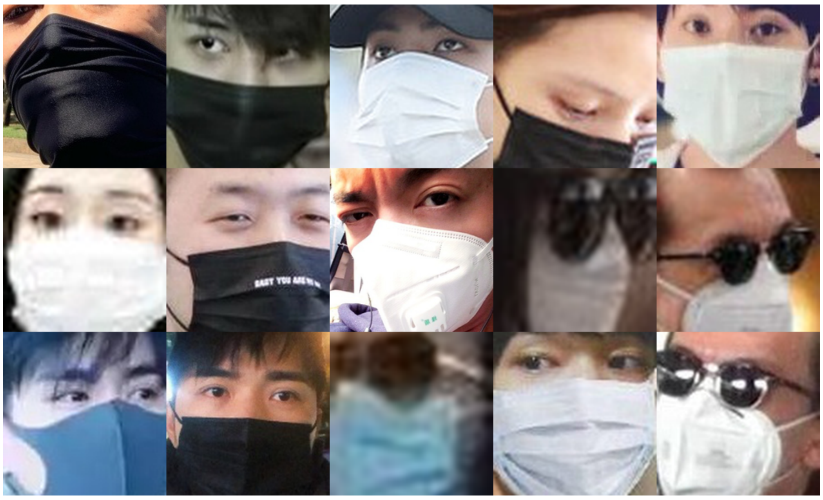


Figure 3.11: 15 exemplary images where MTCNN could not detect the person.

RMFD [224]. 15 randomly selected images where face detection did not work are shown in Fig. 3.11. In contrast, Fig. 3.12 shows 15 randomly selected images where the face detection was successful.

After manually inspecting the cases where face detection did not work, we found an interesting behavior of MTCNN. Fig. 3.10 shows a masked person wearing eyeglasses. MTCNN detects the reflected person in both lenses while missing the person wearing the eyeglasses. This behavior raises the question whether MTCNN ever detects a person if it has already detected a person in its subarea. Therefore, the following experiment was conducted: Two images were manually constructed, each one featuring a person. Without any modification (left-hand side of both Fig. 3.13a and Fig. 3.13b) the person is detected. After inserting another image inside the (fore-)head (Fig. 3.13a) and inside the cheek (Fig. 3.13b)), MTCNN is not able to detect the original person anymore.

While this previously unknown behavior seems somewhat logical, it has a severe potential for abuse: face recognition relying on MTCNN for face detection can easily be evaded by intelligent placement of the image in a face, leading to the actual face staying undiscovered and unrecognized. Furthermore, as shown in Fig. 3.10, MTCNN can be fooled if sunglasses reflect another face. This behav-

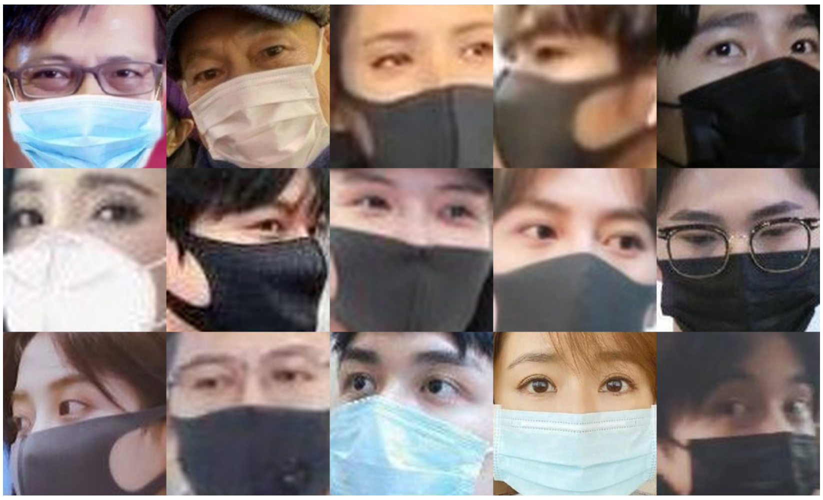


Figure 3.12: 15 exemplary images where MTCNN could detect the person.

ior is particularly problematic since popular tools like Arcface and SphereFace explicitly recommend using MTCNN.

Summary

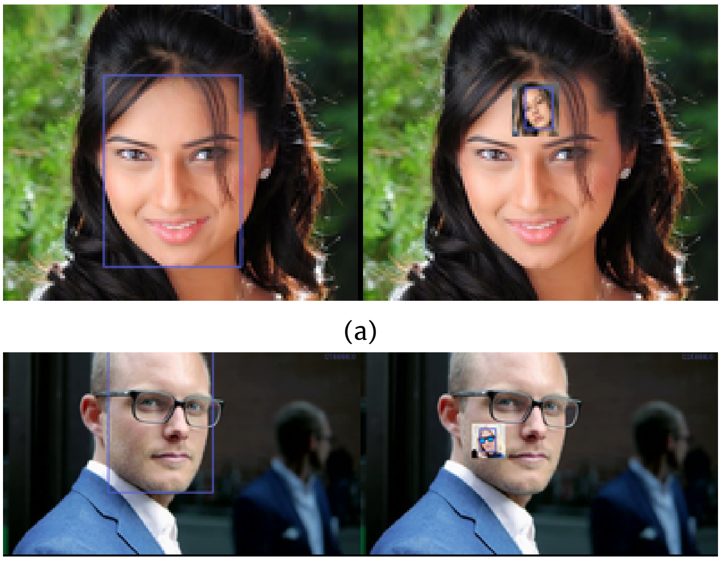
This chapter begins by detailing the state-of-the-art face pipeline (Section 3.1). Retinaface and Arcface offer the highest accuracy, thus they will be used as default models in this thesis and also for our prototype described in Chapter 8. However, these two models are also the slowest, highlighting the need for tailored solutions based on application requirements, which we will do in Chapter 7.

Next, the chapter explores heuristics to enhance preprocessing steps in face recognition pipelines. These include metrics like eye distance relative to face width and eye-mouth distance relative to face height. In our Prototype (Chapter 8) we use these findings and exclude all sensings where the eye distance is less than 8 % of the total face width and where the distance of the eyes to the mouth is less than 20 % of the face height Implementing these heuristics preemptively filters out poor-quality images, thereby improving overall system accuracy and efficiency (Section 3.2).

The chapter also analyzes the performance of three state-of-the-art face detection algorithms on occluded faces. Two different types of occlusions have been studied:

1. automatically modified versions of the CFP dataset, removing various parts of the face, and
2. real world images of people wearing masks.

The region around the nose plays an important role in face detection. Even though all three analyzed face detection algorithms achieve roughly the same accuracy on a dataset without occlusions, Retinaface outperforms both MTCNN and DLIB on most datasets where large parts of the face are missing.



(a)

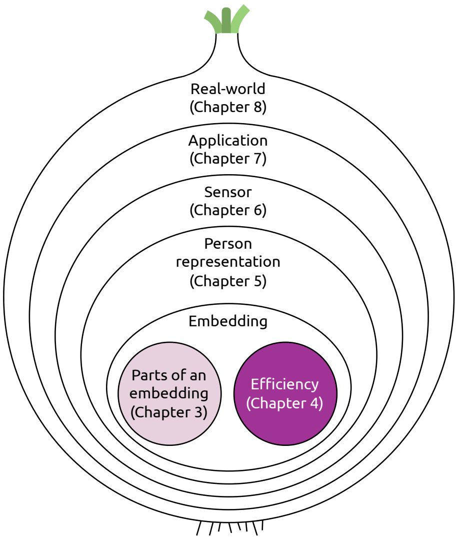
(b)

Figure 3.13: MTCNN detects the original person (left-hand side in a) and b)). If another person is inserted inside the head (right-hand side in a) and b)), the original person is no longer detected.

Furthermore, this work found an interesting behavior of the popular face detection algorithm MTCNN: If a face is visible inside another face, the larger face will not be detected by MTCNN. This can significantly impact face recognition, which relies on MTCNN for face detection, such as state-of-the-art algorithms such as Arcface and SphereFace.

Chapter 4

Shrinking giants: The power of tiny embeddings



Foundation

The foundation of this chapter is the following paper:
Hofer, Philipp, Philipp Schwarz, Michael Roland, and René Mayrhofer. 2024. Shrinking embeddings, not accuracy: Performance-Preserving Reduction of Facial Embeddings for Complex Face Verification Computations. In *14th International Conference on Pattern Recognition Systems (ICPRS 2024)*. IEEE, London, UK, (July 2024)

In the previous chapter, we focused on gaining a semantic understanding of existing facial embeddings. We dissected the components constituting an embedding and explored the significance of individual facial features. This analysis provided insights into the information encapsulated within these high-dimensional vectors, highlighting their role in accurately representing biometric data. Building on this foundation, we now focus on reducing the size of these embeddings.

Conventional embeddings employed in facial verification systems typically consist of hundreds of floating-point numbers. This widely accepted design paradigm primarily stems from the swift computation of vector distance metrics for identification and authentication, such as the L2 norm. However, high-dimensional embeddings can become a concern when integrated into complex comparative strategies, such as multi-party computations. In this chapter, we challenge the presumption that larger embedding sizes are always superior and provide a comprehensive analysis of the effects and implications of substantially reducing the dimensions of these embeddings (by a factor of 29). We demonstrate that this dramatic size reduction incurs only a minimal compromise in the quality-performance trade-off. This discovery could lead to enhancements in computation efficiency without sacrificing system performance, potentially opening avenues for more sophisticated and decentral uses of facial verification technology.

Source code

To enable other researchers to validate and build upon our findings, the Rust code used in this chapter has been made publicly accessible and can be found at https://github.com/mobilesec/reduced-embeddings-analysis-icprs and in Chapter A.

Why do state-of-the-art face recognition systems use embeddings in the first place? They use embeddings due to the embeddings' ability to efficiently and effectively handle large datasets without needing to retrain for each new face. Unlike earlier systems which required re-training for every new face, embedding-based systems generalize well across different conditions and populations by learning from a diverse initial dataset. This generalization allows for simple and quick integration of new faces by adding their embeddings to the system's database, avoiding the computational burden of retraining. Additionally, these embeddings facilitate rapid, on-the-fly comparisons, enhancing scalability and flexibility in deployment across varied platforms. State-of-the-art facial verification algorithms typically employ high-dimensional floating-point values for their embeddings:

- Deep face recognition [158] (2015): 4,096 dimensions
- VGGFace2 [27] (2017): 2,048 dimensions
- Arcface [52] (2019): 512 dimensions
- SphereFace [123] (2017): 512 dimensions
- AdaFace [106] (2017): 512 dimensions
- FaceNet [181] (2015): 128 dimensions

These high-dimensional embeddings have proven incredibly useful in facial verification and recognition systems. Using numerous floating-point numbers optimizes verification accuracy and ensures high computational efficiency, contributing to their broad acceptance as an industry standard.

Despite the unparalleled accuracy of these embeddings in state-of-the-art facial verification systems, there is a growing motivation to reduce their size for three primary advantages:

1. Reduced-size embeddings significantly enhance multi-party computation capabilities. Systems like Funshade [95] efficiently calculate whether the distance between two embeddings is below a threshold without revealing the actual embeddings, ensuring privacy and efficiency.

2. Improved transmission efficiency, especially in environments not reliant on traditional TCP connections. Specifically, embeddings compact enough to fit within a 509-byte Tor cell [92] can be transmitted more swiftly. Furthermore, the necessity for embeddings to be small enough for inclusion in modified Tor introduction packets, as detailed by recent research [89], highlights their importance in scenarios with strict data size constraints. Consequently, smaller embeddings offer significant data transfer speed and efficiency advantages, particularly beneficial in settings with limited bandwidth or data capacity.
3. Reduced storage requirements, which is especially beneficial for contexts with limited space, such as smart cards. Smaller embeddings allow for more efficient space utilization and increase storage capacity, enhancing device utility and application scope.

We investigate how reducing the embedding size affects facial verification system performance, focusing on the trade-offs between efficiency, privacy, and accuracy. We aim to provide a detailed understanding of the practical implications of optimizing embedding sizes for better computational efficiency and system performance. We challenge the common belief that larger embedding sizes always yield better results in facial verification systems by significantly reducing these dimensions.

Our hypothesis suggests that drastically reducing the embedding size may not proportionally decrease performance, but could enhance computational efficiency. This could allow for more complex comparison functions, such as multi-party computations, potentially improving the decentralization of biometric systems.

In our investigation, we explore two options for embedding reduction:

1. reducing the number of elements within an embedding (dimension reduction) and
2. utilizing smaller data types to represent the elements.

Both strategies come with their inherent advantages and potential drawbacks. Dimension reduction may allow for substantial computational savings, but it may also compromise the richness of the data represented. Using smaller data types can similarly reduce computational demand, yet it raises the concern of losing precision and increasing quantization errors.

The following two sections will focus on each of these approaches in detail. We aim to illuminate the consequences and benefits of these strategies and ultimately determine whether the trade-off between efficiency and performance is viable.

In this chapter, we use two datasets: The portrait-like images from LFW (c.f. Section 2.2.4) and the more challenging CPLFW dataset (c.f. Section 2.2.5).

4.1 Related work

Exploring efficient and compact biometric embeddings is part of the larger field of neural network optimization and model compression. While the specific focus on reducing the size of biometric embeddings is underrepresented in current literature, the extensive research into neural network model minimization offers valuable insights and methodologies that apply to this challenge. This chapter provides a focused summary of selected key techniques in model compression, highlighting their relevance and possible applications in shrinking biometric embeddings. The citations included are representative and not exhaustive, aiming to introduce the most significant and pertinent contributions to this area of study.

Pruning and Sparsity

One of the primary methods in model compression is pruning, which involves removing redundant or non-critical parameters from a neural network. Research by Yang et al. [231] demonstrates a novel approach to enhance neural network efficiency. They introduce a low-cost technique using winners-take-all dropout to regulate dynamic activation sparsity, leading to structured activation sparsity with improved levels. When combined with weight pruning, this method shows significant runtime speedups with minimal accuracy loss, underscoring the potential of pruning in neural network optimization. Furthermore, Shao et al. [189] propose a dynamic scheme for imposing sparse constraints based on filter weights. Their method demonstrates superior pruning performance, substantially reducing parameters and computational costs. These studies collectively highlight the significance of pruning and sparsity in enhancing the efficiency of neural networks, a concept that can be transferred to optimizing biometric embeddings.

Quantization

Quantization, another key technique in model compression, involves reducing the precision of the network's parameters. Marinò et al. [130] explore this concept and propose a novel lossless storage format for CNNs leveraging both weight pruning and quantization. Their findings indicate that such compression techniques can drastically reduce neural networks' space occupancy maintaining competitive performance levels. This approach is directly applicable to biometric embeddings, as it entails representing data with fewer bits, suggesting that lower precision may be sufficient for maintaining the integrity of biometric data.

4.2 Element reduction

Under ideal circumstances, the elements within a biometric embedding would exhibit a balanced distribution, where all elements contribute equally, implying a potential compromise in accuracy should dimensionality reduction occur. This section seeks to understand the impact of reducing these dimensions on model performance.

We begin by evaluating 6,000 test pairs from the LFW dataset using the L2 norm as the distance metric, selected for its widespread use and effectiveness in similar research. Hofer et al.'s suggestion that the choice of distance metric is not crucially supported the decision to use the L2 norm, given its proven efficiency in related empirical studies [87].

To evaluate facial verification models' verification, a threshold is established. Embeddings for face pairs are calculated, and their L2 distance is measured. Pairs are then classified as the same person if the distance is below the threshold or different individuals if above.

We optimized the threshold to minimize both false positives and negatives by systematically testing every threshold value that altered at least one outcome. For example, if the set of distances were 1.0, 1.2, 2.0, we evaluated threshold values as 1.1 (between 1.0 and 1.2), and 1.6 (between 1.2 and 2.0) to ensure comprehensive coverage and precise adjustments.

Tests on three face detection and two verification models showed consistent trends, with the combination of Retinaface and Arcface being the most effective. Therefore, this chapter will employ this combination, which utilizes an embedding comprised of 512 dimensions of 32-bit floating points. This establishes our baseline: These models achieved 99.3 % accuracy on the LFW dataset, using all 512 dimensions, where accuracy is defined as the ratio of correct predictions (true positives and true negatives) to the total number of predictions.

We examined the accuracy impact of using lower-dimensional embeddings by sequentially removing elements and recalculating the error rate and optimal threshold for each reduced dimension. This process continued until a single-dimensional embedding was reached, illustrating the accuracy trade-offs at each reduction stage. Despite its impracticality, a single-dimensional embedding was included to represent the effects of dimensionality variations fully. The outcome of this iterative process is depicted in Fig. 4.1.

The findings indicate an excess in embedding dimensions, with a reduction in embeddings not initially leading to a significant increase in errors, suggesting possible data streamlining without major performance loss. Further robustness checks, involving 100 reruns with randomly selected indices on sets with 7, 32, 120, and 200 dimensions, confirmed the initial observation's consistency across different dimensions (Fig. 4.2), underscoring the likelihood that many facial verification systems operate with unnecessary data. These specific dimensions were selected for further investigation due to their intriguing characteristics observed in the raw data presented in Fig. 4.1. This consistency adds weight to our initial finding: many facial verification systems likely carry more data than necessary.

Some index subsets perform better due to lower error rates, but slight differences among all tested combinations suggest that choosing a specific subset may not significantly affect the outcome. Still, steady performance across 100 random indices does not rule out the possibility of an optimal set.

In order to verify the existence of this optimal set, we identify the subset with the lowest error rate, within our experimental framework, requiring evaluation of all combinations. However, enumerating all $\sum_{n=1}^{512} \binom{n}{512}$ combinations is computationally infeasible.

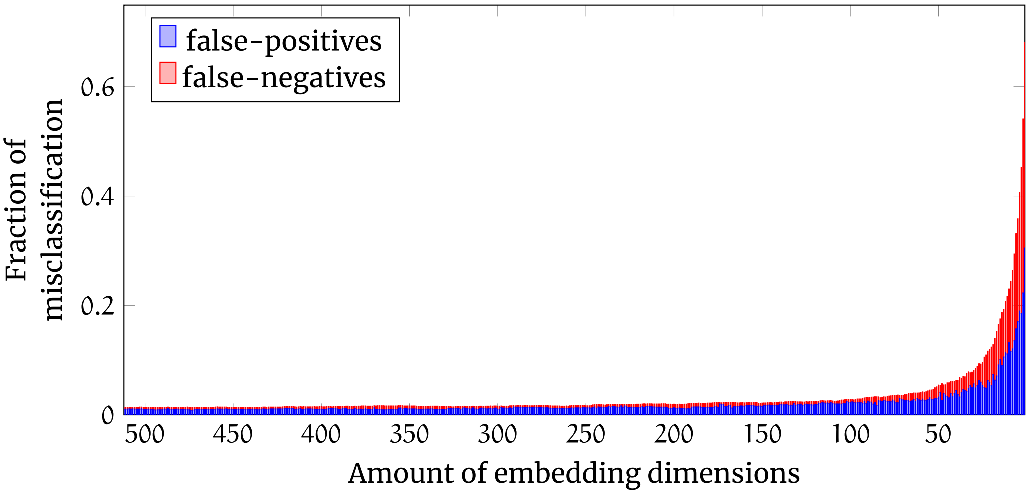


Figure 4.1: The error rate on the LFW dataset correlates with embedding dimensionality, rapidly converging to 40/6000 errors. Using 100-dimensional embeddings results in slightly more errors (69) than the full 512 dimensions (40).

Data analysis in Fig. 4.1 shows that using only the first 32 indices yields a 96.1 % accuracy, close to the 99.3 % accuracy achieved with all 512 indices. This highlights the effectiveness of our simplified model. Guided by these insights, we resolved to scrutinize every possible combination encapsulated within these initial 32 elements. It presents a viable opportunity to conduct an exhaustive exploration while retaining the potential to yield satisfactory accuracy.

We must note that we approached this analysis with a holistic view of all subsets' potential combined performances. The performance of a particular index, for example, index 16 in an initial round, does not necessarily dictate a superior result in subsequent rounds. Two separate indices, despite their individual performances not reaching the same peak as index 16, could in conjunction yield a superior outcome.

To account for these variations in possible performance, reliance on results from prior iterations is avoided. Each new round commences with a fresh, comprehensive exploration of all possible subsets. This approach enables the identification of any advantageous combinations that might otherwise be overlooked if one relies solely on preceding results.

Our consumer-grade hardware[1] processes our pipeline at ~25 iterations per second with a reasonable low power consumption of 15 W. The most effective subset and its corresponding error rate for each iteration are reported in Table 4.1. The seventh iteration took 35 hours, with the eighth and ninth projected to take 116 and 311 hours, respectively. Extrapolating, an exhaustive analysis of all 32-element subsets would take an estimated 5.5 years on this setup.

Our code is yet to be optimized. Dedicated optimization efforts could significantly reduce computation time. A complementary strategy is employing a greedy search instead of a full grid search. A greedy search iteratively adds the

[1] Intel Core i7-10510U

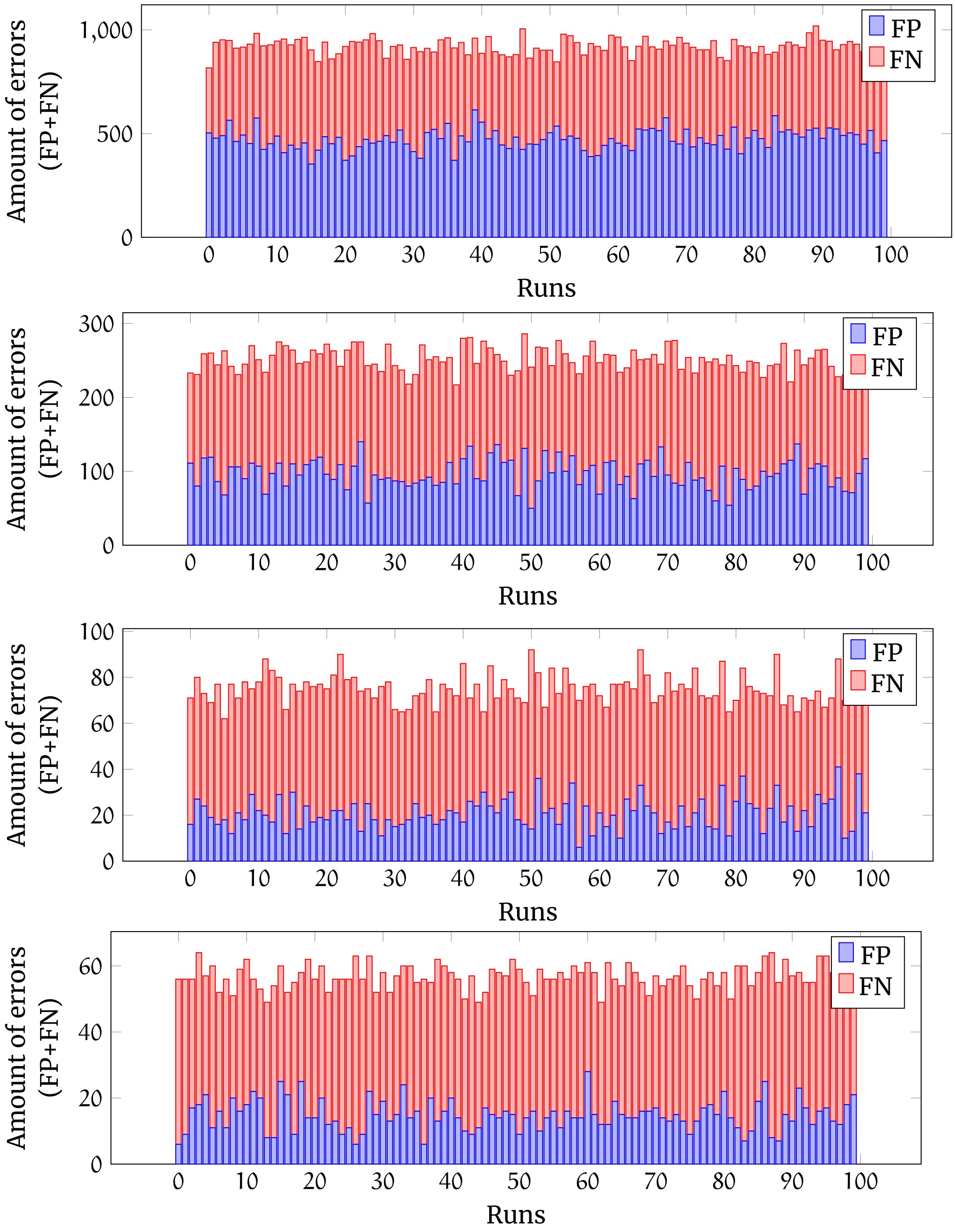


Figure 4.2: Exploring potential index-dependent bias in the LFW dataset is shown, wherein particular index sets might yield significantly superior performance. This is assessed through 100 iterations, with 7, 32, 120, and 200 dimensions (top to bottom) randomly chosen and subsequently tested for error rates.

Table 4.1: Brute-force search of the best elements in the LFW dataset.

Iteration	Elements	Percentage of misclassification
1	16	32.2%
2	16, 31	25.2%
3	16, 25, 31	20.9%
4	16, 25, 29, 31	17.7%
5	14, 16, 28, 29, 31	15.8%
6	14, 15, 16, 28, 29, 31	14.1%
7	1, 3, 16, 17, 24, 26, 31	12.4%

best element to an optimal set, reducing the search space to $32 * n$, where n is the number of elements, enabling completion within minutes (Table 4.2). This non-exhaustive method offers a practical solution with lower computational demand.

We conducted a comparative analysis to evaluate the effectiveness of greedy search compared to exhaustive brute-force search Results for the first seven elements suggest that greedy search can effectively substitute for brute-force search, as shown in Fig. 4.3.

Subsequently, in our follow-up experiment, we utilized the top-performing indices identified by the greedy search instead of the initial elements. This modified approach produced promising results, achieving an accuracy rate of 96.1 % with just 32 elements. This performance is notably close to the original 99.3 % accuracy achieved using all 512 elements, illustrating the potential of the greedy search method in reducing computational load while maintaining a high degree of accuracy.

A significant benefit of the greedy search method is its flexibility, as it is not confined to 32 elements and can instead efficiently evaluate error rates across all 512 dimensions. While the absolute values may be lower, the trend observed in this greedy-search setting closely aligns with that seen in Fig. 4.1. Additionally, it is important to note a slight but noticeable increase in the error rate beyond the 230th index marker. This suggests that the presence of certain elements is detrimental to the performance of face verification. Such an inference reiterates the notion that reducing the embedding size, particularly during the training phase, may even enhance accuracy.

The results of the greedy search reinforce our observations from Fig. 4.1, confirming that a significantly high level of accuracy can be maintained with a reduced subset of elements. To validate the robustness and applicability of our findings, we employed the greedy search method on the more demanding CPLFW dataset, characterized by its array of complexities including unfavorable angles and diverse lighting conditions, as depicted in Fig. 4.6.

While the graphical representation resembles Fig. 4.1, an increased optimal error rate, reflective of the greater complexity inherent in the CPLFW dataset, is observed. However, a meticulous examination reveals that the greedy algorithm selects distinct elements for each dataset. Even so, employing a rank-1

Table 4.2: Greedy search for the best indices set using the first 32 elements.

It.	Elements	# Err
1	16	1934
2	16, 31	1514
3	16, 25, 31	1254
4	16, 25, 29, 31	1062
5	16, 17, 25, 29, 31	950
6	4, 16, 17, 25, 29, 31	853
7	1, 4, 16, 17, 25, 29, 31	773
8	1, 4, 15, 16, 17, 25, 29, 31	686
9	1, 4, 15, 16, 17, 18, 25, 29, 31	636
10	0, 1, 4, 15, 16, 17, 18, 25, 29, 31	583
11	0, 1, 4, 15, 16, 17, 18, 25, 29, 30, 31	542
12	0, 1, 4, 10, 15, 16, 17, 18, 25, 29, 30, 31	511
13	0, 1, 4, 10, 15, 16, 17, 18, 24, 25, 29, 30, 31	477
14	0, 1, 4, 10, 12, 15, 16, 17, 18, 24, 25, 29, 30, 31	440
15	0, 1, 4, 10, 12, 15, 16, 17, 18, 20, 24, 25, 29, 30, 31	408
16	0, 1, 4, 10, 12, 15, 16, 17, 18, 20, 24, 25, 26, 29, 30, 31	386
17	0, 1, 4, 9, 10, 12, 15, 16, 17, 18, 20, 24, 25, 26, 29, 30, 31	369
18	0, 1, 4, 9, 10, 12, 15, 16, 17, 18, 20, 21, 24, 25, 26, 29, 30, 31	355
19	0, 1, 4, 9, 10, 12, 14, 15, 16, 17, 18, 20, 21, 24, 25, 26, 29, 30, 31	343
20	0, 1, 2, 4, 9, 10, 12, 14, 15, 16, 17, 18, 20, 21, 24, 25, 26, 29, 30, 31	329
21	0, 1, 2, 4, 7, 9, 10, 12, 14, 15, 16, 17, 18, 20, 21, 24, 25, 26, 29, 30, 31	318
22	0, 1, 2, 4, 7, 9, 10, 11, 12, 14, 15, 16, 17, 18, 20, 21, 24, 25, 26, 29, 30, 31	303
23	0, 1, 2, 4, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 20, 21, 24, 25, 26, 29, 30, 31	297
24	0, 1, 2, 4, 6, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 20, 21, 24, 25, 26, 29, 30, 31	281
25	0, 1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 20, 21, 24, 25, 26, 29, 30, 31	267
26	0, 1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 20, 21, 24, 25, 26, 27, 29, 30, 31	255
27	0, 1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 20, 21, 24, 25, 26, 27, 28, 29, 30, 31	248
28	0, 1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 19, 20, 21, 24, 25, 26, 27, 28, 29, 30, 31	239
29	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 19, 20, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31	233
30	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31	234
31	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31	225
32	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31	233

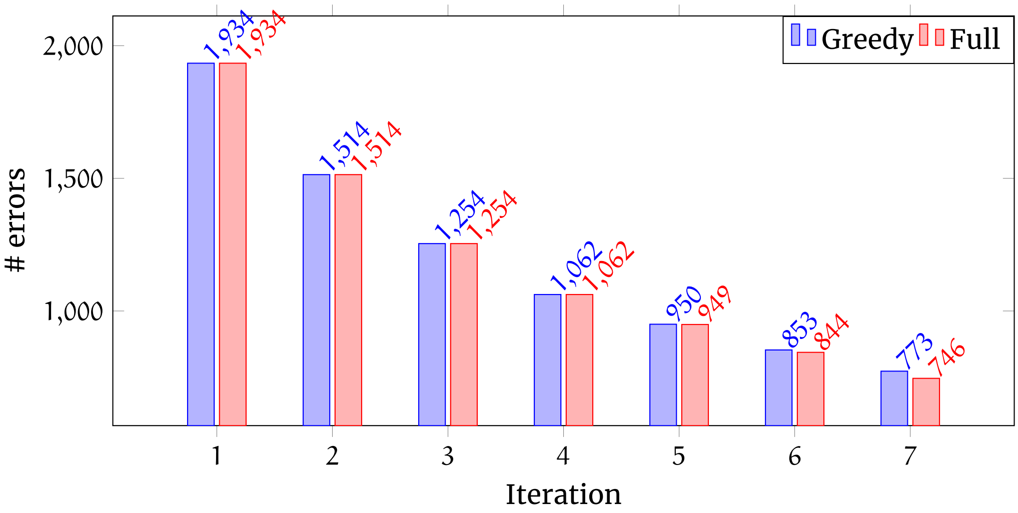


Figure 4.3: In the LFW dataset, error rates obtained from the greedy search are compared with those from the exhaustive brute-force search. The first four elements align perfectly; thereafter, the performance begins to exhibit a slight decline. Nevertheless, the similar shape suggests that the greedy search is a satisfactory proxy.

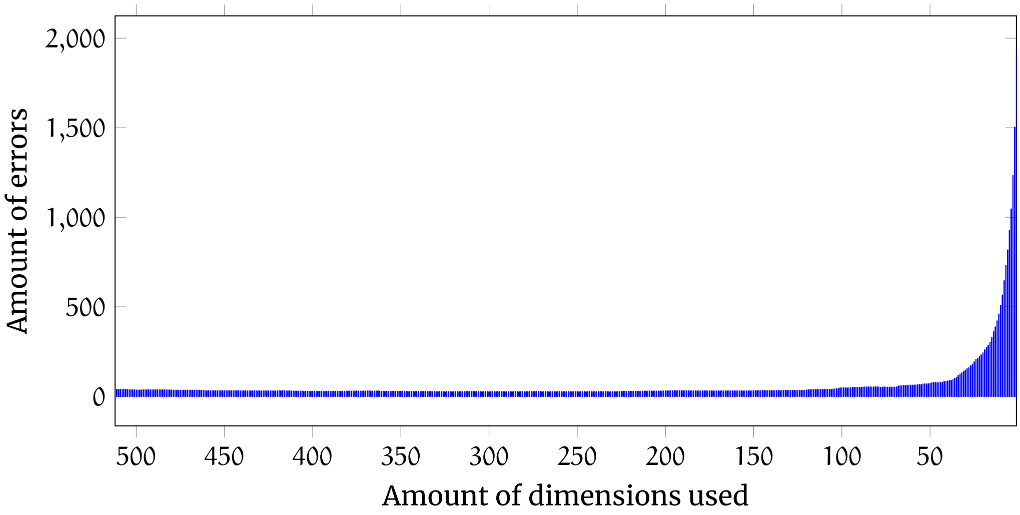


Figure 4.4: Greedy search over all 512 dimensions on the LFW dataset.

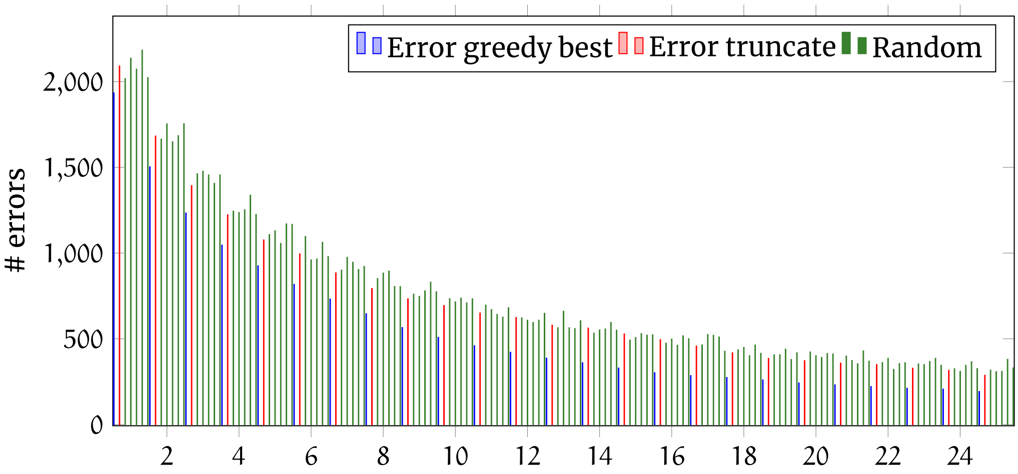


Figure 4.5: Comparative analysis of the greedy search against our other configurations (initial and random elements). X-axis: Amount of dimensions used

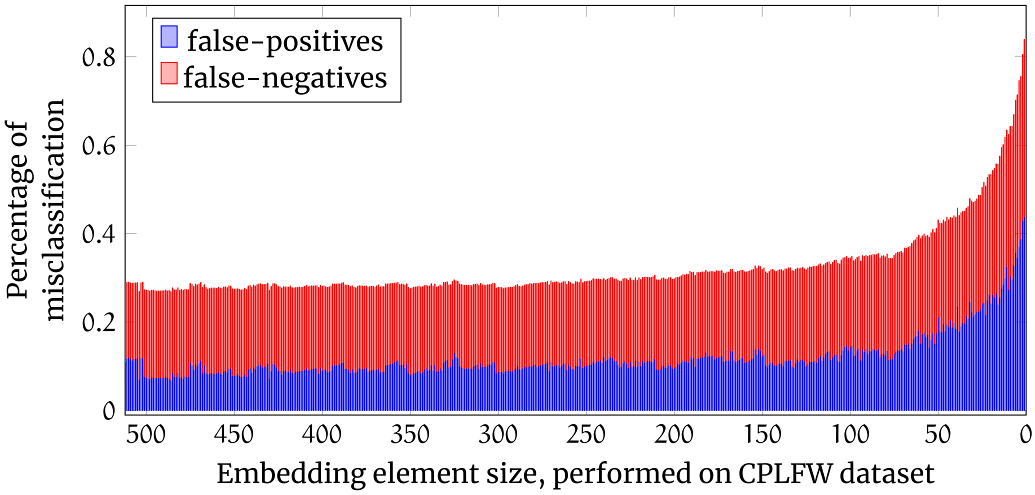


Figure 4.6: The shape of the error rate on the CPLFW dataset (shown here) is similar to the error rate of LFW dataset (Fig. 4.1).

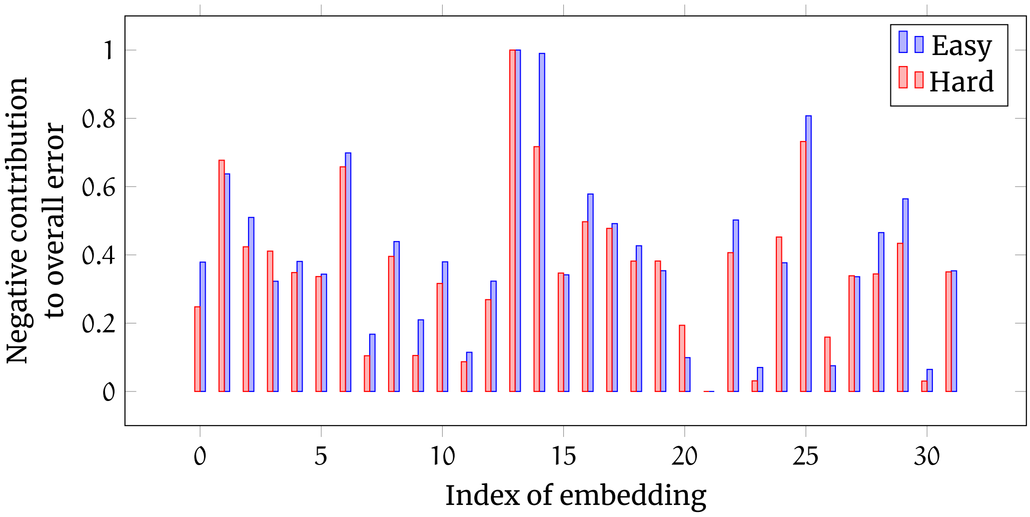


Figure 4.7: Cross-dataset index evaluation: Analysis of index contributions to overall classification error in L2 distance metrics. A more significant difference between the two bars signifies higher classification inaccuracies related to a specific index. The blue bar represents a visualization of the LFW dataset, whereas the CPLFW dataset is depicted in the red bar. Notably, the index contributions to the total distance exhibit striking similarities across both datasets.

approach reveals that the top-performing index from the LFW dataset can still deliver substantial results on the CPLFW, albeit not necessarily as the foremost choice.

The impact of each index on the resulting L2 distance is evaluated to evaluate their cross-dataset applicability. For indices increasing the error in classifying two images of the same individual (indicating a misclassification), the associated distance increases the index value. Conversely, for indices that err in distinguishing between different individuals (indicating a correct classification), the index value is reduced by the corresponding distance. Thus, a higher value of a particular index reflects its contribution to an increased overall classification error, demonstrating its propensity to introduce “wrongness” into the total distance metric. Finally, all index values are normalized to a range between 0 and 1 to standardize the results and enable a balanced comparison. The distribution of the first 32 indices is depicted in Fig. 4.7, providing insights into their respective influence on classification accuracy.

Interestingly, and contrary to initial expectations, the indices that minimally contribute to the error differ from those identified by the greedy search. For instance, despite index 16’s superior performance in the greedy search, it is ranked among the least effective in the heatmap. This discrepancy could be explained by a uniform contribution of these indices to the total error, rendering selecting a specific index less crucial.

To further validate these observations and compare the relative effectiveness of various configurations, we assessed the greedy search’s performance relative to other methods, including the use of initial elements and random selection. The comparative results are detailed in Fig. 4.5. Moreover, an examination of

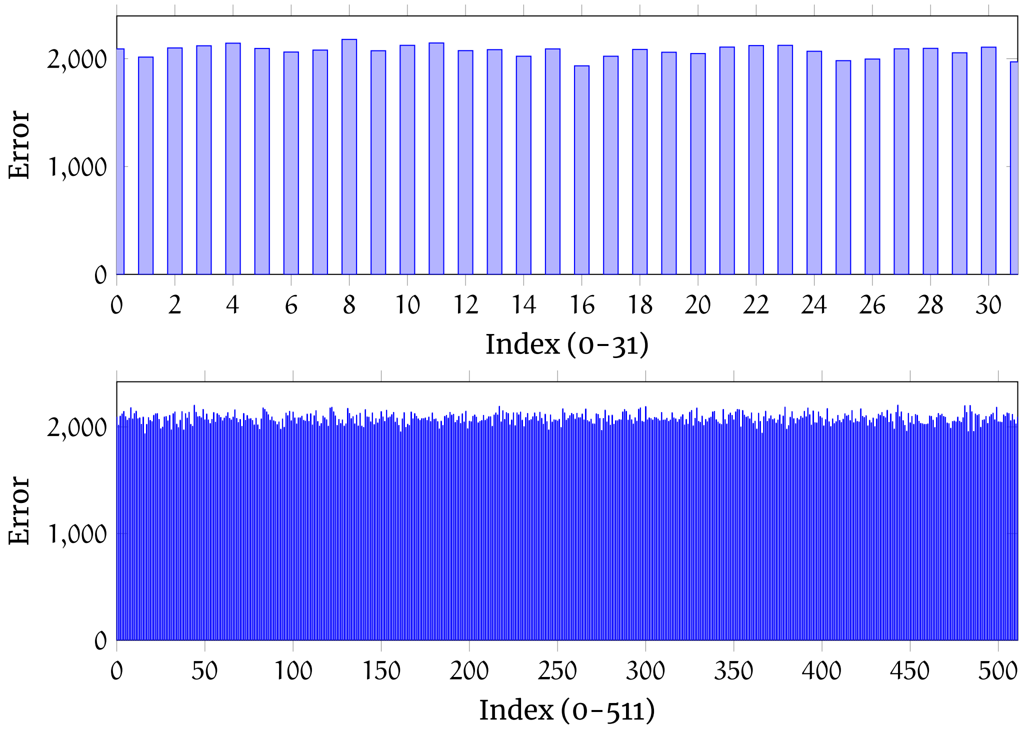


Figure 4.8: Visual representation of the error generated when only the corresponding index is utilized.

each index's individual contribution shows that the choice of the starting index has a negligible impact on the overall error.

4.3 Data quantization

As the field of machine learning evolves, researchers are exploring efficient ways to compress and optimize models. Data quantization is a key technique that reduces data size and complexity by mapping inputs to fewer outputs, improving storage and processing efficiency with little impact on performance. The concept of data quantization is established in machine learning. Liang et al. [117] showed that quantization might not significantly affect system performance. Our study differs by focusing on output quantization, specifically of facial embeddings. This section will cover quantization techniques, their impact on facial verification systems, and their implementation to minimize facial embedding size with minimal performance loss.

This study aligns with the element reduction approach outlined in Section 4.2, targeting the optimization of facial embeddings through distinct methods. Element reduction eliminates redundant dimensions, whereas data quantization enhances value representation efficiency. In our approach, we assume uniform quantization for the embeddings, where each level of quantization uniformly represents a segment of the input data range. This simplifies the complexity and ensures more predictable effects on system performance. Section 4.4 examines their synergistic potential to efficiently minimize facial embeddings

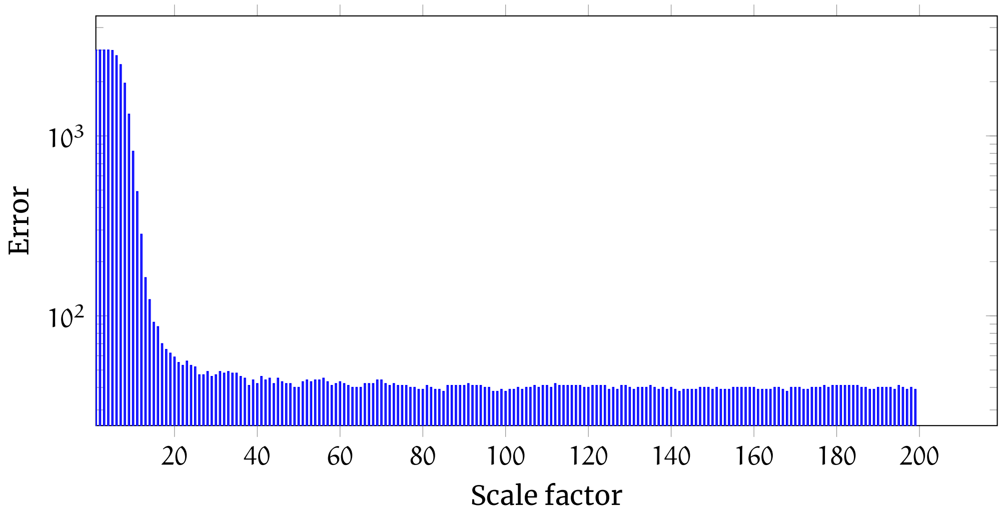


Figure 4.9: Visualization of different scale factors. The optimal threshold, which minimizes the combined rate of False Positives (FP) and False Negatives (FN), is dynamically recalculated for each respective scale factor.

without compromising verification system quality and performance.

Quantization of facial embeddings involves converting the 32-bit floating-point values to alternative data types. This study assesses the impact of such conversions on error rates, emphasizing the importance of precision beyond the optimal fixed point range in filter design, as small differences might have a significant impact, especially if they are near the threshold range. The goal is to analyze the balance between data compactness and accuracy.

The original 32-bit floating point values fall within a relatively constrained range: approximately -0.2 to $+0.2$. We employ calibration to investigate the contribution of the float mantissa and exponent to the overall information conveyed. We multiply the original values by a range of factors (between 1 and 200) and subsequently, convert the scaled values into an integer format (which incurs loss of information).

Fig. 4.9 illustrates the relationship between each scaling factor and the corresponding error rate, providing a detailed overview of the quantization impact.

Analysis of error rates across different scaling factors reveals a plateau effect commencing at a calibration value of 70. Beyond this point, we observe no significant decline in the error rate, as evidenced by a marginal difference in verification accuracy (99.32 % as opposed to the original 99.33 %).

The range of the resulting values with a scale factor of 70 spans from -19 to 21. This suggests the adequacy of a 6-bit signed integer datatype for representing our data post-scaling.

An alternative approach involves adjusting the scaled values with an offset of +19, thus repositioning the range to span from 0 to 40. Consequently, the values can be efficiently represented using an unsigned 6-bit integer datatype.

4.4 Proposed pipeline

We suggest a pipeline that operates on only 70 indices determined by the heatmap of each index over the LFW dataset (which can be found in *final_indices.txt* in the accompanying Git repository) and cast the embeddings to an 8-bit integer format (as there is no 6-bit integer data type in most programming languages) with a calibration value of 70.

This modification leads to a reduction of over 29 times the bit requirement (from 16,384 bits to 560 bits), with only a slight decrease in accuracy (99.3 % to 98.6 %), corresponding to a net increase of 44 errors (out of 6,000 comparisons).

When evaluated on the challenging CPLFW dataset, we observe a slightly larger reduction in accuracy (from 85.4 % to 79.87 %), resulting in an increase of 331 errors (out of 5,964 comparisons).

Nonetheless, this approach preserves the computational efficiency, exhibiting a theoretical reduction in computation by a factor of 29.

Given that we are using only 6 of the 8 available bits, we could apply a more practical approach to leverage the standard 8-bit hardware platforms. By encoding the 70 sets of 6 bits into approximately 53 sets of 8 bits, we optimize the use of existing storage and computational capacity. The adjustment could enhance storage efficiency up to a factor of 38 (424 bits / 16,384 bits), while preserving the precision of the results. This strategy enables us to make more efficient use of hardware capabilities without compromising the accuracy of our computations.

An additional advantage of this compact size is its compatibility with an SHA-512 hash function due to similar sizes, presenting potential benefits for specific applications. For instance, cryptographic algorithms operating with such data can readily use these embeddings without requiring any modifications.

4.5 Practical implications of compact embeddings

Beyond the sheer academic fascination and the computational benefits lies the real-world applicability of these compact facial embeddings. Given the current trends towards decentralized and edge computing, the size reduction becomes even more paramount. Edge devices, such as smartphones or embedded devices, often have limited computational power and storage capacities compared to large GPU clusters. We can deploy facial verification capabilities on these devices without overburdening them by employing compact embeddings.

Furthermore, smaller embeddings imply faster encryption and decryption processes in security and data privacy. If facial verification data needs to be transmitted over a network, compact embeddings mean fewer data to send, resulting in quicker transmission times and reduced chances of interception.

Summary

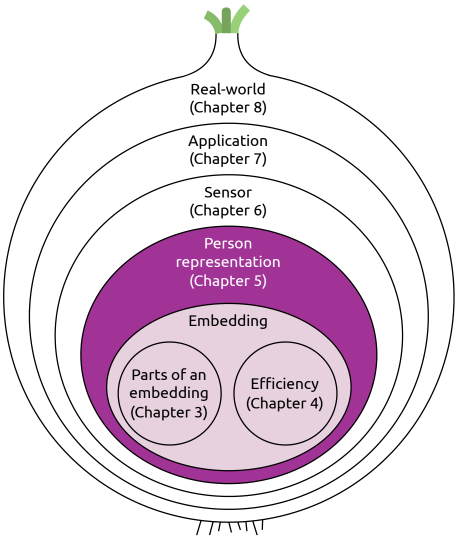
We conducted a thorough investigation into the effects of reducing embedding size on the accuracy of facial verification algorithms, specifically by decreasing the number of elements and modifying data types. Contrary to the common belief that high-dimensional embeddings are crucial for maintaining accuracy, our findings reveal that accuracy levels above 90 % can be achieved even with a substantial reduction in embedding size—approximately by a factor of 29. This discovery has significant implications for applications in environments with limited computational power and storage capacity.

The reduced size of the embeddings not only improves efficiency in comparison, storage, and transmission but also makes multi-party computation (MPC) possible in the context of facial verification. Smaller embeddings enable more efficient comparisons in complex facial recognition tasks, which is particularly beneficial for decentralized and privacy-sensitive systems. The compact nature of these embeddings also reduces storage requirements, making them ideal for use in devices with limited storage capacity, such as smart cards. Furthermore, the smaller size facilitates faster data transmission over networks, which is crucial in scenarios with restricted bandwidth or where rapid data transfer is essential.

In summary, our work demonstrates that it is possible to achieve high accuracy in facial verification while significantly reducing embedding size, thereby enabling more efficient and practical applications, including the feasibility of MPC in real-world settings with constrained resources.

Chapter 5

One template to rule them all: Fusing embeddings



Foundation

The foundation of this chapter is the following paper...
Hofer, Philipp, Michael Roland, Philipp Schwarz, and René Mayrhofer. 2023. Efficient Aggregation of Face Embeddings for Decentralized Face Recognition Deployments. In *Proceedings of the 9th International Conference on Information Systems Security and Privacy (ICISSP 2023)*. SciTePress, Lisbon, Portugal, (February 2023), pp. 279–286. DOI: 10.5220/0011599300003405
... and its extended version in a journal:
Hofer, Philipp, Michael Roland, René Mayrhofer, and Philipp Schwarz. 2023. Optimizing Distributed Face Recognition Systems through Efficient Aggregation of Facial Embeddings. *Advances in Artificial Intelligence and Machine Learning*, 3, 1, (February 2023), 693–711. DOI: 10.54364/AAIML.2023.1146

This chapter builds upon the foundations laid in previous discussions on the detailed exploration of embedding contents (Chapter 3) and embedding minimization (Chapter 4). Our journey so far has shown that while individual embeddings offer a snapshot of an identity, real-world applications benefit from a more comprehensive representation: This chapter introduces the concept of fusing embeddings to respond to this requirement, improving efficiency and privacy in decentralized systems. By aggregating multiple embeddings, we can construct a template that better represents an individual's biometric data.

In centralized systems, the current live embedding is typically compared to a set of template images of the person using fast distance functions such as L2 norm or cosine similarity. This process is inefficient on two levels:

1. Requiring multiple (on a holistic system point of view, redundant) similarity calculations would hurt both provider diversification and hinder small providers in serving larger quantities of users because of increased hardware requirements. Ideally, one could combine different aspects of these multiple embeddings extracted from face images with as little data as possible. Since research on still image face recognition is extensive, and an embedded camera sensor device can often derive embeddings of the currently visible person online, creating a new, aggregated embedding based on all images available of an individual would not change the backbone of state-of-the-art face recognition pipelines.

2. Having a single (aggregated) embedding, thus not depending on multiple similarity computations, minimizes network traffic, which is especially significant for decentralized, embedded systems.

Performing many distance calculations is possible due to the straightforward nature of the comparisons within a centralized database. However, in decentralized systems, the scenario changes significantly. Sensors should avoid sending the current live embedding to all potentially relevant devices where a template of that person might be stored. Conversely, the device storing the template should avoid leaking its data to the sensor. This creates a privacy and security challenge.

Multi-party computation (MPC) techniques, such as Funshade [95], can be employed to address this issue. MPC allows the comparison of embeddings without revealing the actual data, thus preserving privacy. The result of the MPC computation is a boolean value indicating whether the distance between the two embeddings is smaller than a predefined threshold. However, performing this calculation within MPC is not trivial. Comparing multiple template embeddings to the current live embedding can be complex, resulting in significant time overhead, including potential network requests. To overcome this challenge, we propose reducing multiple template embeddings to a single fused embedding. This reduction aims to streamline the process, making performing MPC efficiently in decentralized networks feasible. By consolidating the templates into one embedding, we can maintain MPC's security and privacy advantages while mitigating the time complexity associated with multiple comparisons.

The fusion process involves statistical techniques that integrate multiple embeddings while preserving the critical features necessary for identification. This method was optimized through extensive testing on our new in-the-wild dataset, which provided real-world conditions essential for validating our approach. Implementing this fusion technique allows for scalable and efficient biometric systems that respect user privacy while offering the flexibility needed in decentralized frameworks. It reduces network load, which is crucial for systems operating in bandwidth-constrained environments. It supports a user-centric model where individuals can manage their biometric templates without relying on a central authority.

This chapter evaluates different methods of aggregating face embeddings from efficiency and accuracy perspectives (Section 5.2). We also examine the limit of sufficient image quantity, analyzing whether there is a clear point at which adding more images does not significantly enhance recognition accuracy. We propose a new in-the-wild dataset to validate whether using multiple images in different settings significantly boosts accuracy. Subjects take around 50 images of themselves in a single setting, which only takes about 3 seconds, ensuring practical usability. Additional images in diverse settings are used to approximate the true embedding and verify performance improvements.

5.1 Multi-image face recognition

In order to evaluate and compare different face recognition models, they are tested against public datasets. Many of these face recognition datasets typically have two properties:

1. High quality: As the datasets are created with training face recognition models in mind, the images of a person mainly consist of a portrait image in a fairly high resolution.
2. High quantity of people: Typically, neural network bias becomes less when more images are used. Therefore, datasets strive for a high amount of images.

Most datasets define a fixed set of pairs of images to allow for objective evaluation of face recognition methods. This strategy uses a single image as a template in state-of-the-art face recognition pipelines. This template is then compared with positive (same person) and negative (different person) matches. This approach tests a critical metric of face recognition: How well it performs on still images. Compared to more complex scenarios, only testing on still images is efficient at runtime, which decreases computation time to evaluate the accuracy of a dataset. However, there are different aspects this method does not test, such as how to handle multiple images or even video streams of a person.

In reality, these ignored aspects are essential, as live images from cameras do not produce high-quality images similar to those from many available and commonly used face recognition datasets. Instead, the person-camera angle is far from optimal. The person is not directly in front of the camera; thus, the face is quite small. Furthermore, the face can be occluded, e.g., with a scarf, sunglasses, or hair. In these real-world settings, face recognition pipelines have more difficulty recognizing people than with public datasets, although new datasets try to represent these challenges. Nevertheless, there is a potential benefit of real-world scenarios: Many images of a single person are available, as the person is presumably visible for (at least) many seconds, and thus, a camera is able to capture significantly more than one image.

One way of bridging the gap of having multiple images of the same person and being of lower quality is to merge the embeddings obtained from multiple images into a single embedding. More accurate templates, by definition, lead to accuracy improvements in face recognition. The idea behind using multiple images is that it is not possible to capture a perfect representation of a face in a single picture for various reasons:

- The image only captures part of the face. Covering frontal and profile pictures in a single 2D image is technically impossible.
- External conditions, such as lighting, camera quality, and insolation, change.
- People themselves change over time: Growing a beard, getting wrinkles or a new hair cut, putting on makeup, getting pierced or tattooed, getting a scar, or having surgical interventions.
- Different accessories, such as (sun) glasses, headgear, earrings, or masks are worn.

While a single image cannot account for all these different settings, multiple images can capture different face areas and settings. Therefore, using multiple images provides more information about the individual's face, and we therefore expect increased accuracy. As introduced in Section 1, comparing the current live image with multiple embeddings of the same person is unfavorable in some situations due to hardware and network constraints. For an efficient face recognition pipeline, it would be best to have only a single embedding that is used as a template for a person. This would allow the system to make use of the vast literature on single-face image recognition.

In contrast to this single-embedding approach, in recent years, other work has been published in the domain of video face recognition [71, 124, 169, 171, 251]. Most of these papers propose an additional neural network to perform the weighting of different embeddings [71, 124, 171, 230]. These additional networks have a significant runtime impact, especially on embedded devices, as they need to perform an additional inference step. In order to be runtime-efficient even on embedded hardware, this chapter focuses on creating a single embedding.

In state-of-the-art face recognition tools, embeddings are high-dimensional vectors. If multiple embeddings should be aggregated into a single one, this opens up the following questions:

Q1. How do we (numerically) *best* aggregate the embeddings, and is this aggregation increasing face recognition performance? How can we define *best*?

Q2. After knowing how to aggregate embeddings, how many images are necessary and useful? Is there a point from which adding additional embeddings do not significantly increase accuracy?

Depending on the application, there may or may not be a lot of data available for each person. Therefore, in many situations (e.g. user enrollment) it could make the process significantly easier if the data can be recorded in a single session and therefore feature only one setting. This leads to the question:

Q3. Is it beneficial to use different settings? Is it worth creating images with and without (typical) accessories, such as face masks, glasses, and scarves?

Similarly, expecting many images from a new user in various settings might be unrealistic. Verifying that these different settings belong to the same person makes it even more complicated. It is easy and practical to capture a couple of images in one place.

Q4. Is it enough to use only images while we rotate our heads for the aggregated embedding, similar to the process of how some smartphones enroll users' faces? Is the accuracy increased if we include totally different settings in the aggregated embedding?

5.2 Embedding aggregation

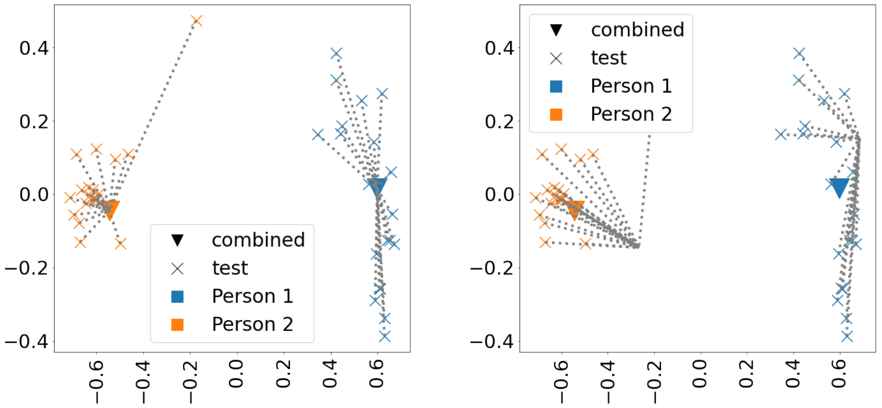
This chapter evaluates different aggregation strategies and proposes efficient ways of aggregating embeddings in order to create a single, efficient template-embedding containing as much information as possible. If multiple images of a person are used, the position of the person of interest has to be extracted in each frame. These positions could be either fed to a neural network that expects multiple images (or a video) as input (*video-based face recognition*) or the embedding could be extracted for each frame and then aggregated (*imageset-based face recognition*).

Video face recognition networks have to perform all necessary steps in a single network. Extracting the embeddings frame-by-frame and only then aggregating these embeddings to a single template allows for a much more modular pipeline. This goes hand in hand with traditional face recognition pipeline approaches, which can be separated into face detection, face tracking, and face recognition, and therefore also allow for individual optimization of each part. Additionally, systems using this approach can use its vast literature, as the field of (still) image face recognition is much more advanced than video face recognition. Therefore, in this chapter, we focus on the modular approach of extracting embeddings from every image and then aggregating them.

Since we want to aggregate multiple embeddings extracted from single frames, we need an *aggregation strategy*. Literature typically calculates the mean of each dimension of the embedding, e.g. as proposed by Deng et al. [52]:

$$\begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix}, \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} \to \begin{pmatrix} \text{mean}(a_1, b_1) \\ \vdots \\ \text{mean}(a_n, b_n) \end{pmatrix}$$

Fig. 5.1 shows the instance space of two people. The x- and y-axis represent the PCR-reduced form of their embeddings. The triangle represents the average of each dimension of the embedding for each person. There is no analysis of whether it is useful to use the mean of each dimension, or whether there are better approaches to aggregate embeddings. There is not even an analysis if calculating the mean of the embeddings improves the accuracy of face



(a) the mean-template.

(b) the first image.

Figure 5.1: Distance of the embeddings to...

Figure 5.2: Example images of a person from the CelebA dataset.

recognition pipelines. In order to verify this hypothesis, a baseline is needed to compare the performance of aggregated embeddings to. In this chapter, building on the motivation outlined in Section 3.1.3, we employ pre-trained state-of-the-art models for face detection and recognition: Retinaface [50] and Arcface [52], respectively. Arcface receives a single image as input and creates a 512-dimensional vector. Even though we did not explicitly test different architectures, we expect similar results on semantically similar networks.

5.3 Dataset adaptation

We selected the CelebA dataset [125] for evaluating the face recognition models due to its high number of images per individual to meet the requirements of our analysis. Furthermore, this dataset comprises a large number of images across thousands of individuals, making it particularly suitable for robust evaluation. A comprehensive description of the dataset is provided in Section 2.2.6. Notably, out of the 10,177 individuals included in the dataset, 2,343 have exactly 30 images each.

In order to reduce the chance of having outliers, we remove all images with less than 30 images (Fig. 5.3). Furthermore, we cleaned the dataset by performing

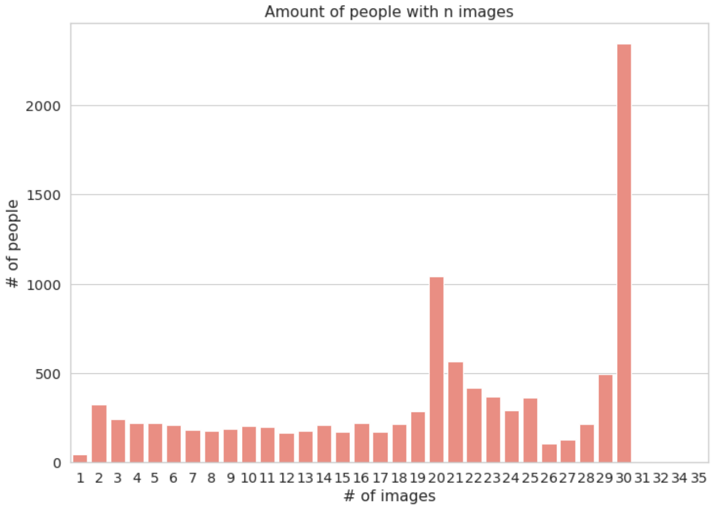


Figure 5.3: Distribution of number of people concerning their number of images.

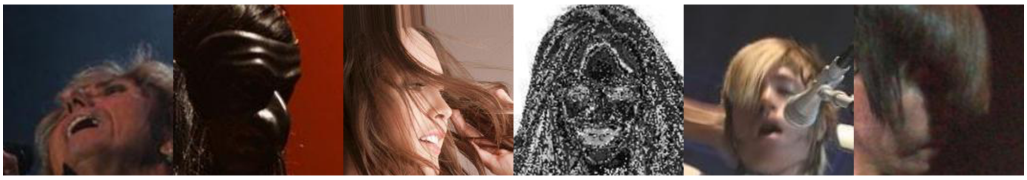


Figure 5.4: Example images where Retinaface could not detect a person.

face detection with Retinaface. From the initial 2.343 people with 30 images, there are 20 people containing an image where face detection could not detect a face—mainly due to too much occlusion. 18 randomly chosen images where face detection did not work are shown in Fig. 5.4. To ensure a consistent dataset, we removed all images of these 20 people, resulting in a final set of 2.323 people.

The CelebA dataset has been pre-processed so that the main person is in the center of the image. If multiple faces are detected, we take the most central person and ignore the remaining ones.

There are 30 images of 2.323 people each in our cleaned dataset, resulting in a total number of 69.690 images. In order to objectively evaluate the difference between different aggregation strategies, we reserve 10 random images of each person as potential template images. Since the dataset does not have a specific order, without loss of generality and for reproducibility, we reserve the first 10 images as potential template images.

In the first setting (*baseline*), we (only) use the embedding of the first image and ignore images 2–9:

$$\text{template}_{\text{baseline}}(\text{person}) = \text{emb}_{\text{person}}[0]. \tag{5.1}$$

Calculating the mean of different embeddings is only one possible strategy to aggregate embeddings. For different methods, such as taking the minimum of each dimension, we numerically aggregate each dimension of the embeddings from images 1–9, using its respective aggregation strategy (as):

$$\begin{pmatrix} e_{11} \\ e_{12} \\ \vdots \\ e_{1m} \end{pmatrix}, \dots, \begin{pmatrix} e_{n1} \\ e_{n2} \\ \vdots \\ e_{nm} \end{pmatrix} \rightarrow \begin{pmatrix} \mathrm{as}(e_{11}, \dots, e_{n1}) \\ \mathrm{as}(e_{12}, \dots, e_{n2}) \\ \vdots \\ \mathrm{as}(e_{1m}, \dots, e_{nm}) \end{pmatrix}. \tag{5.2}$$

Typically, face recognition models are trained such that the L2 distance between two faces represents their (non-)similarity:

$$\mathrm{dist}(\mathrm{emb}_1, \mathrm{emb}_2) = \sqrt{\sum_{i=1}^{n} (\mathrm{emb}_1 - \mathrm{emb}_2)^2}. \tag{5.3}$$

Applications set a threshold for their specific task, under which two faces are recognized as the same person. This decision is based on their safety requirements. For security-critical applications, the threshold should be lowered, which results in fewer false-positives (but potentially more false-negatives).

$$\begin{aligned} &\mathrm{isSamePerson}(\mathrm{emb}_1, \mathrm{emb}_2) = \\ &\begin{cases} 1, & \text{if } \mathrm{dist}(\mathrm{emb}_1, \mathrm{emb}_2) \leq \textbf{threshold} \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \tag{5.4}$$

In order to compare the strategies, we take the average distance of the template to each embedding from images 11–30 as our metric:

$$\mathrm{err} = \frac{\sum_{p \in \mathrm{people}} \sum_{\mathrm{emb}_{\mathrm{test}} \in \mathrm{testEmbs}} \mathrm{dist}(\mathrm{emb}_{\mathrm{test}}, p_{\mathrm{template}})}{\mathrm{len}(\mathrm{people}) \times \mathrm{len}(\mathrm{testEmbs})}$$

A smaller *error* represents a higher confidence of the network that the template belongs to the test images. Semantically, the error specifies the template's average distance (Eq. 5.3) to each test image.

The *CelebA* row in Table 5.1 shows the resulting distance between the template- and test-embeddings. Fig. 5.5 visually represents the CelebA column. With respect to our Q1: Except for the (cheating) *optimal* setting (which we discuss later), and the *Best template per comp* (where we employ an even greater degree of bias by utilizing the optimal reference image for each comparison), the best aggregation strategy is using the mean of every dimension of the embedding. As the distance compared to the baseline is significantly lower in the *average* (and *median*) setting, this clearly shows the practical impact of using multiple (in our case 10) images as templates. Aggregating multiple embeddings using the mean significantly outperforms the baseline. Fig. 5.1 shows the intuition behind this behavior on the first two people. If more than one image of a person is used, the resulting embedding more accurately approximates the optimal embedding. The average of these test embeddings represents the optimal

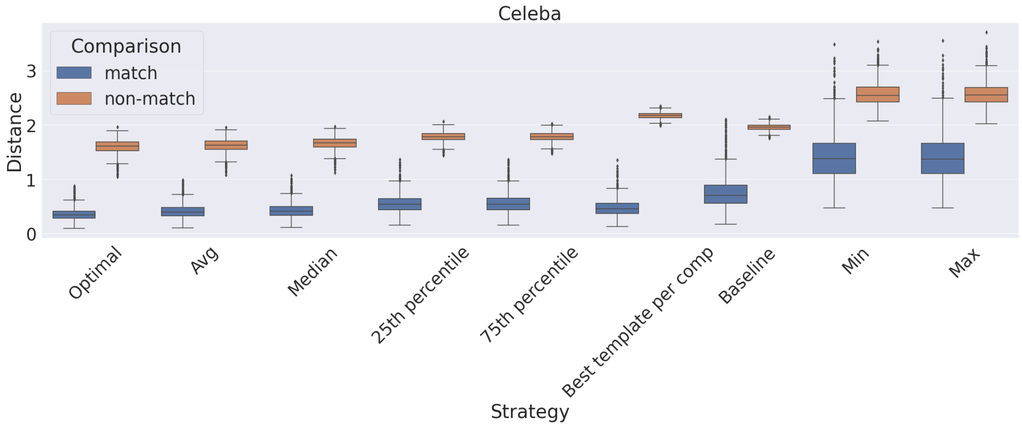


Figure 5.5: Average distance of template- to test-embeddings in CelebA dataset.

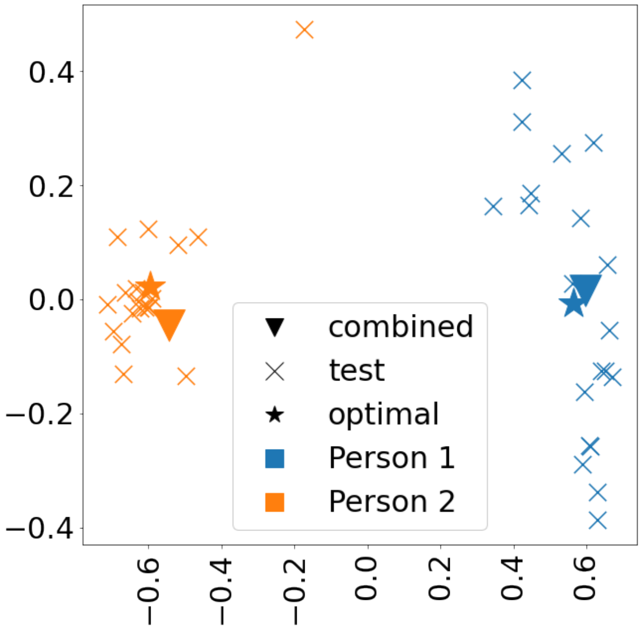


Figure 5.6: Instance space of optimal vs average aggregation.

Table 5.1: This table shows the average distances of the template embedding to the test embeddings with respect to different aggregation strategies. The value in brackets represents the factor of the distance of that strategy compared to the baseline. A factor of 2 means that the average distance of the baseline is twice as high as this particular aggregation strategy. For *match*es, a higher factor is favorable, while for *non-match*es, a lower factor is better. The gray rows are displayed for comparison reasons only, as they cheat and use information not available in production.

		CelebA	Panshot-Normal	Panshot-Smile
Baseline	Match	0.748 (1.0x)	0.617 (1.0x)	0.612 (1.0x)
	Non-Match	1.958 (1.0x)	1.888 (1.0x)	1.840 (1.0x)
Avg	Match	0.410 (1.8x)	0.404 (1.5x)	0.425 (1.4x)
	Non-Match	1.622 (1.2x)	1.520 (1.2x)	1.476 (1.2x)
Median	Match	0.422 (1.8x)	0.423 (1.5x)	0.443 (1.4x)
	Non-Match	1.662 (1.2x)	1.611 (1.2x)	1.556 (1.2x)
Min	Match	1.414 (0.5x)	2.208 (0.3x)	2.160 (0.3x)
	Non-Match	2.567 (0.8x)	3.107 (0.6x)	3.006 (0.6x)
Max	Match	1.409 (0.5x)	2.186 (0.3x)	2.166 (0.3x)
	Non-Match	2.564 (0.8x)	3.080 (0.6x)	3.010 (0.6x)
25th percentile	Match	0.552 (1.4x)	0.557 (1.1x)	0.574 (1.1x)
	Non-Match	1.781 (1.1x)	1.718 (1.1x)	1.661 (1.1x)
75th percentile	Match	0.552 (1.4x)	0.560 (1.1x)	0.581 (1.1x)
	Non-Match	1.781 (1.1x)	1.716 (1.1x)	1.668 (1.1x)
Optimal	Match	0.354 (2.1x)	0.383 (1.6x)	0.382 (1.6x)
	Non-Match	1.604 (1.2x)	1.501 (1.3x)	1.443 (1.3x)
Best template per comp	Match	0.471 (1.6x)	0.170 (3.6x)	0.214 (2.9x)
	Non-Match	2.173 (0.9x)	2.195 (0.9x)	2.121 (0.9x)

embedding for an individual with respect to the current test images, as it minimizes the respective distance.

So far, it has been shown that using an average of 10 images significantly outperforms using a single image as a template. Naturally, the question arises of whether the accuracy will still increase if more images are used. Is there a limit above which additional images will not further improve accuracy (Q2)?

We need a dataset with more images of the same person to answer this question. For this purpose, we used the LFW dataset [94] as it contains hundreds of images of the same people. In particular, we use the 5 people in the LFW dataset who have more than 100 images. Two randomly selected images of these 5 people are shown in Figure 5.7.

For each person, the embedding of the first image serves as the starting point. Next, the embedding of the second image is extracted. The first point of each plot in Fig. 5.8 represents the sum of the difference between these two embeddings. We then combine all previously used embeddings into our template. Afterward, we extract the embedding of the following image, calculate its difference from the template, and plot the value. We continue with this approach until we used every available image.

Interestingly, it looks like a (fuzzy) inverse log function. Intuitively, this makes sense as new images initially contain a lot of new information, but after the template consists of many aggregated images, a new image cannot provide as much new information as in the beginning. Furthermore, there seems to be a limit of roughly 50 images, after which the embedding does not change significantly anymore. Another aspect to point out is that the graph has some *upticks*. After looking at the specific images that cause these effects, we see that they all present a new variation of the face (either a new face angle or different accessories).

Section 5.2 used images of the same person in different settings, such as different hairstyles, lighting, and location. The dataset mainly consists of frontal images, with the person looking directly into the camera. Some modern smartphones provide the ability to unlock the phone by rotating the phone around the head. This is probably used not only to detect the person's liveness but also to

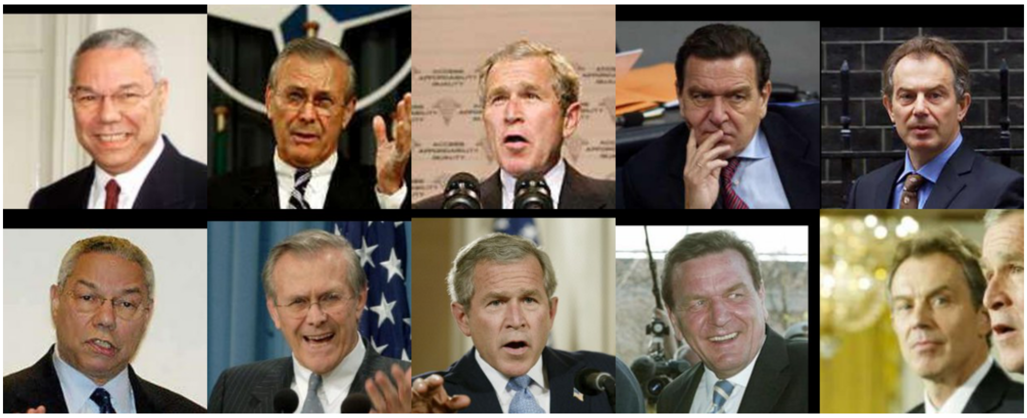


Figure 5.7: 2 exemplary images of the LFW dataset of the 5 people with more than 100 images.

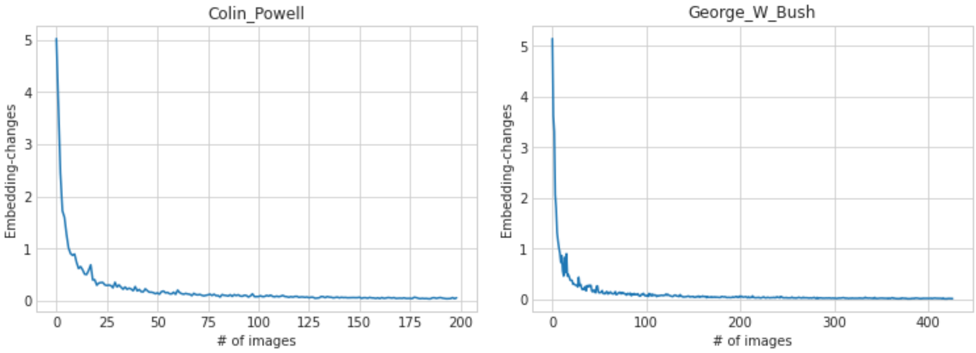


Figure 5.8: Numeric embedding differences shown for 2 people from the LFW dataset.

increase the amount of information gained from the camera. Is the difference in angle from this type of recording enough to utilize the benefit of combining embeddings discussed so far (Q3)?

Therefore, we did a similar analysis on a different dataset: Pan Shot Face Database (PSFD) [64]. This dataset features 30 participants from 9 perspectives. Every perspective contains 5 *look directions* (straight, slightly top left, slightly top right, slightly bottom right, and slightly bottom left) and 4 distinct *facial expressions* (normal, smiling, eyes closed, and mouth slightly opened). This gives us 5,400 images to work with.

For the first test, we used all images with a *normal* face expression as a template and evaluated its average distance to all other images. The result is visible in the *PS-Normal* row in Table 5.1.

People in this dataset are easier recognized compared to the CelebA dataset, which is reflected in a lower average distance (Table 5.1). For the CelebA dataset, the template which consists of 10 images is performing 1.8 times better than if only a single image is used as template. Interestingly, this improvement is in the same order of magnitude on our new dataset: 1.5 times better.

Images are professional portrait photographs (e.g., used as profile images) of the subject to simulate real-world templates. In the second scenario, templates are created with images of the smiling person. The outcome of this *PS-Smile* setting is not significantly different from the original *PS-Normal* setting (c.f. Table 5.1). Thus, it does not make a significant difference in the person's facial expression while creating template images.

5.4 Single setting performance

In our experiments so far, we used images of the same people in different settings, as these are the most common images provided by available datasets. In practice, however, it is convenient for both the provider and the individual to only use images taken at the time of physical enrollment. The provider would benefit by ensuring that the individual is not spoofing the system, e.g., by using images from other people [1]—which would break security guarantees for all kinds of authentication systems, both with publicly issued credentials such as passports and with accounts enrolled with only a single (e.g., building access control) system. The advantage for the user is better usability, as they do not have to provide any additional data besides their participation in the enrollment procedure. With enrollment interaction limited to a few seconds, we argue that creating a more diverse set of input face images to improve recognition accuracy as proposed in this chapter takes less effort than creating a traditional user account by setting a new password.

Unfortunately, there are no publicly available datasets that systematically contain both images of people in the same setting (e.g., only rotating the head, as performed for some mobile phone face authentication implementations) and also images in different settings. In order to test our hypothesis of only using a single setting while additional images of the same person in different settings do not increase accuracy, we created a new dataset, which we called *In-The-Wild Face Angle Dataset*. We will also use this dataset to answer Q4. Inspired by Datasheets for Datasets [68], we describe the dataset on our website: digidow.eu/experiments/face-angle-dataset.

Dataset

> To facilitate further research, this dataset is available as open source. However, due to legal and ethical considerations, access is granted only after signing an agreement. The agreement and further information can be found at digidow.eu/experiments/face-angle-dataset

In order to test the increased performance if multiple images recorded in a single session are used, we calculated a rolling average of the template images. The first data point for each person is equal to the first embedding. The second data point is the average of the first two embeddings. The last data point is the average of all embeddings of this particular person.

In order to quantify the performance, the 10 images of each person in different settings are not used as template images. Instead, the average distance between the rolling average of the template images and test images is calculated.

The average distance between each person's first image and their test images is, on average, 0.699. If we use not only the first image but the average of all template images, the average distance drops significantly to 0.291. Fig. 5.9 shows the average plot of a person.

Interestingly, this opposes the previous findings as the distance grows smaller even after the limit of roughly 50 images. We argue that this is due to the fact that not the amount of images, but the amount of **semantically different** images are important.

[1] Note that providing wrong or even specifically manufactured images to the enrollment process could have multiple goals: in addition to the apparent enrollment of a set of images containing faces of two or multiple persons to make them all recognized as a single system user, malicious users might try to attack the embedding computation or matching approaches directly by exploiting model weaknesses through specifically tampered input. The exact attack vector is outside the scope of this chapter, as our proposed collection of multiple images in a single, controlled enrollment session is assumed to prevent both targets at once.

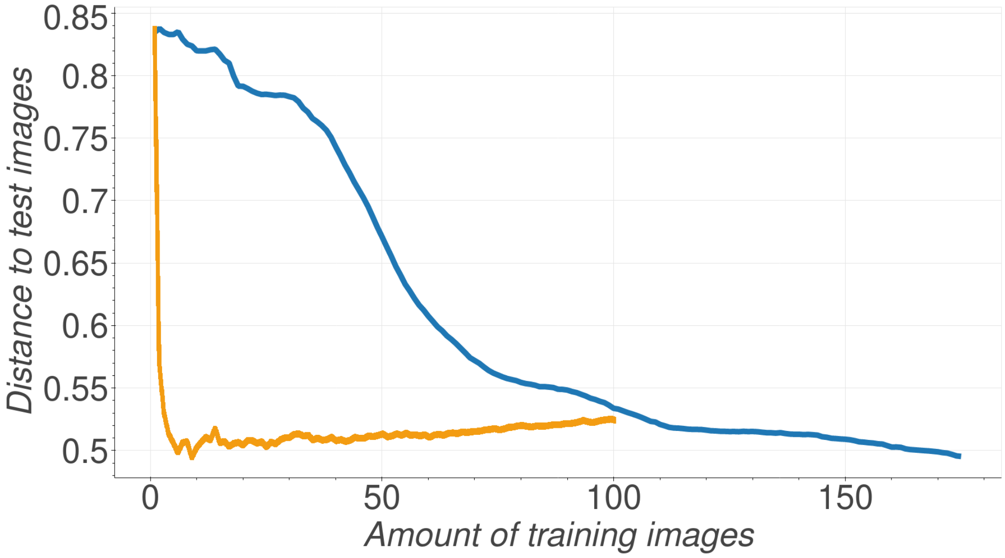


Figure 5.9: Rolling distance average of the aggregated embedding to the test images. The y-axis shows the average distance to the test images (orange → greedy search; blue → ordered).

Table 5.2: This table shows the average distance if only a subsection of the training images are used.

Used images	Distance
All images (117-463)	0.291
Every 10th image (11-46)	0.294
Every 20th image (5-23)	0.297
Every 50th image (2-9)	0.325
1 image	0.699

To verify this, we ran the same experiment but used only every n^{th} image as a template image. The results are shown in Table 5.2: The distance decreases if more images are used (all images: 0.291, $n = 10$: 0.294, $n = 20$: 0.297, $n = 50$: 0.325, single image: 0.699). However, distance improvements are certainly not linear and are leveling off at some point. The improvement from using just a few images is only marginally better than using hundreds of images, suggesting that the number of images plays a minor role.

If the improvement best seen in Fig. 5.5 is due to having different face angles, we expect a similar improvement if we switch from using dozens to just a few images picturing different angles. Therefore, instead of using the template images in sequence, a greedy search on every iteration should result in the best embedding for each step. At every step, we create the new average embedding for all remaining images of the person, calculate the new distance to the test images, and select the one that minimizes this distance. Table 5.3 shows that after using just the 3 best images, accuracy already improved significantly, and there is little room for improvement (0.315 for the 3 best images vs 0.291 if all

Table 5.3: This table shows the average distance if images have been selected greedily.

After n-th best images	1	2	3
Avg. distance	0.571	0.370	0.315

images are used). After manually inspecting the top images for each person, in 82 % of the cases, the first 3 images are one frontal image and two profile images from each side. Further work could add convergence criteria to select the best amount of images automatically.

5.5 Related work

Chowdhury et al. [35] proposed an interesting change: Instead of using the mean-weighting of features, they propose to use the maximum instead. This should reduce the overfit on dominant angles and generalize better [35]. However, this finding could not be replicated with this dataset, as the *minimum* and *maximum* settings perform significantly worse than the baseline (c.f. Table 5.1). One potential cause for this bad performance is that outliers have too much impact on the final template. Therefore, we created another template by using $\{25, 75\}\mathrm{th}$ quantile of each dimension of the embedding, which scores significantly better than both the *minimum* and *maximum* setting, but not as well as the *average* aggregation strategy.

Rao et al. [169] created a pipeline with a similar goal. Instead of aggregating the embeddings into a single template, they created a neural network that receives raw images as input. As the networks have full access to the whole image (instead of an embedding only), this approach offers the possibility of higher accuracy at the drastic expense of runtime performance and is thus not really suitable for embedded systems.

Furthermore, in the last years, much effort is spent on deciding how to weigh different dimensions of embeddings [124, 171, 230]. Even though some of these approaches look promising, they are not ideal for embedded systems, as most of them use additional hardware-intense computations. Therefore, this work does not favor any specific image over another.

Balsdon et al. [13] showed that the accuracy of humans doing face identification significantly improves in a "wisdom of crowd" setting compared to individual's performance. This indicates, that a similar effect is demonstrable if a system combines embeddings not only from a single face recognition neural network but also from multiple different ones. Therefore, further work could use the proposed method of combining embeddings of different neural networks, potentially using the same aggregation strategies as analyzed in the present chapter.

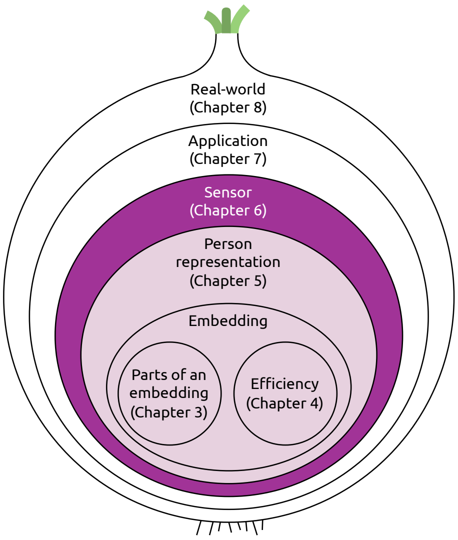
Summary

In this work, we evaluated different aggregation strategies, concluding that aggregating embeddings by taking the average of each dimension provides the highest improvement in accuracy while remaining compatible with state-of-the-art face recognition pipelines as already widely deployed in the field. We stress that this was one of the design goals of our work. Our results indicate that such improvements can be directly applied to existing (embedded and distributed) systems with changes to only the enrollment and template computation processes, but not the live recognition pipelines.

Even though some previous work implicitly used this average aggregation strategy, its effectiveness has not been evaluated. We base this proposal on an extensive evaluation of different aggregation strategies using both different public datasets and creating a new dataset which is publicly available for research purposes. After quantitatively analyzing the number of images used to generate templates, we find that it only plays a minor role, while different perspectives—we refer to them as semantically different input—significantly improve the performance of face recognition pipelines. For an efficient, decentralized system, we propose using (just) 3 images per template: one frontal image and one from each side. These images may share the same setting; thus, if there is a physical enrollment, these images can be taken live. This increases both the system's correctness (as there are fewer options to spoof it) and its usability (as the user does not have to provide larger sets of images or even video footage).

Chapter 6

The speed of sight: Optimizing face detection for embedded systems



Foundation

The foundation of this chapter is the following paper:
Hofer, Philipp, Philipp Schwarz, Michael Roland, and René Mayrhofer. 2023. Face to Face with Efficiency: Real-Time Face Recognition Pipelines on Embedded Devices. In *21st International Conference on Advances in Mobile Computing & Multimedia Intelligence (MoMM 2023)*. ACM, Bali, Indonesia, (December 2023)

Building upon the foundations laid in earlier chapters, which focused on understanding, optimizing, and combining facial feature embeddings, this chapter transitions to addressing the practical challenges and solutions for implementing these optimized systems on embedded hardware. This progression from theoretical optimization to practical application is essential for ensuring the viability of face recognition systems in real-world, resource-constrained environments.

Real-time face recognition on decentralized systems and embedded hardware presents numerous challenges, with the primary issue being the trade-off between accuracy and inference-time on constrained hardware resources. Achieving higher accuracy often results in longer inference times, which can be impractical for applications requiring quick responses. Therefore, optimizing this trade-off is crucial for the feasibility of real-time applications.

To address this challenge, we first conduct a comparative study on different face recognition distance functions and introduce an inference-time/accuracy plot. This plot provides a clear visual representation to facilitate the comparison of different face recognition models. It helps to identify the optimal balance between inference-time and accuracy based on specific application needs.

Building on these insights, we propose a combination of multiple models with distinct characteristics. This approach leverages each model's strengths while mitigating its individual weaknesses, thereby optimizing performance for diverse application requirements. The integration of these models aims to achieve a balance of accuracy, reliability, and speed.

We demonstrate the practicality of our approach by developing a multimodel face recognition pipeline. This pipeline utilizes two face detection models positioned at opposite ends of the inference-time/accuracy spectrum. By strategically integrating these models on an embedded device, we achieve a balance where the more accurate model is used only when necessary, and the faster model is employed for generating quick proposals. This method improves the trade-off between inference-time and accuracy, providing a practical guideline for developing real-time face recognition systems on embedded devices.

6.1 Intricacies of SOTA face pipelines

Authenticating a person using biometrics requires two main steps: face detection, followed by face recognition (c.f. Section 3.1). In this section, we iterate over state-of-the-art models to improve time performance (on embedded systems). First, the system must accurately detect and locate the face within the image or video frame. Once the face is detected, the system can then extract the relevant facial features necessary for recognition. Recognition involves comparing these features to a database of known faces to determine the individual's identity. Therefore, accurate detection and recognition are essential for effective and reliable biometric authentication.

To quantify the performance of our face detection models, we employed the LFW dataset, which provides a diverse and challenging set of facial images that are well-suited for benchmarking recognition accuracy and robustness. A detailed description of this dataset can be found in Section 2.2.4. For evaluation, we adopted a metric where a predicted bounding box is considered successful if it overlaps by more than 50 % with the ground truth bounding box. This threshold was chosen due to its widespread acceptance and effectiveness in accurately reflecting model performance, as evidenced in studies such as Yang et al. [232]. By utilizing this established metric, we ensured a robust and consistent quantitative assessment, facilitating a reliable accuracy comparison across different face detection models. This methodological choice aligns with standard practices in the field, enhancing our results' validity and comparability.

6.1.1 Face detection

With the increasing demand for facial recognition technology, a wide variety of face detection models have been developed. Each model has certain advantages over their competitors: Some focus on finding tiny faces [115], occluded faces [111], or using multiple camera angles [61].

In order to quantify the quality of networks and be able to compare different models, they are evaluated on publicly available datasets. There is a focus on accuracy: Wider Face [232] shows a precision-recall curve, LFW [159] shows the ROC-curve and the corresponding area under the curve, VGGFace2 [27] shows false(-positive)-acceptance-rates and rank-accuracies, UMD Faces [14] shows the normalized mean error.

In this section, we will provide a brief overview of (four) popular choices of face detection networks.

Retinaface

Retinaface is based on a single-shot detector framework and uses a fully convolutional neural network (FCN) to detect faces in images. The architecture of Retinaface consists of three main components: a backbone network, a multi-scale feature pyramid network, and three task-specific heads.

The backbone network is responsible for feature extraction and is typically a pre-trained ResNet or MobileNet. The feature pyramid network then takes the feature maps generated by the backbone network and produces a set of multi-scale feature maps. Finally, the task-specific heads, consisting of a classification head, a regression head, and a landmark head, are applied to each feature map to predict the presence of a face, its bounding box, and its facial landmarks.

ULFGFD

ULFGFD is specifically designed to be lightweight and suitable for deployment on edge computing devices. The small size, just over 1 MB, stands out in particular. The network is based on a single-shot detector (SSD) architecture and consists of a backbone and prediction networks. The backbone network is a lightweight MobileNetV2 architecture that is used to extract features from input images. The prediction network consists of a set of convolutional layers that are used to predict the bounding boxes and confidence scores of faces in the input images.

ULFGFD also uses a feature pyramid network (FPN) to detect faces at different scales. The FPN consists of a set of convolutional layers that are used to generate feature maps at different resolutions. These feature maps are then used to predict the bounding boxes and confidence scores of faces at different scales.

YuNet

is a deep neural network architecture designed for efficient face detection and recognition in real-world scenarios [62].

YuNet is composed of three main components: a lightweight backbone network, a feature pyramid network (FPN), and a detection head. The backbone network is based on MobileNetV2, a popular architecture known for its efficiency and low computational cost.

The detection head of YuNet is responsible for predicting the locations of faces in the input image. It consists of a set of convolutional layers followed by two parallel branches. One branch performs classification to determine whether a given region of the image contains a face or not, while the other branch performs regression to predict the face's bounding box coordinates.

Haarcascade

is a widely used computer vision algorithm for face detection, having been introduced by Viola and Jones as early as 2001 [217]. Despite being around for over two decades, Haarcascade remains a popular choice for face detection in various applications due to its simplicity, efficiency, and effectiveness.

The Haarcascade algorithm works by using a series of classifiers to detect faces within an image. Each classifier is composed of a set of weak learners, which are typically decision trees that evaluate simple features such as edges and corners. These features are calculated on a sliding window that moves across the image, with the goal of detecting faces at different scales and orientations.

One limitation of Haarcascade is that it can be sensitive to changes in lighting conditions and occlusion, which can result in false positives or missed detections.

6.1.2 Face recognition

Face recognition is the process of identifying an individual based on their distinctive facial features. In recent years, the accuracy, reliability, and efficiency of this process have increased significantly due to advancements in deep learning algorithms and the availability of large datasets.

The majority of state-of-the-art (SOTA) algorithms requires a pre-processed RGB image as input, which is then used to create a high-dimensional vector that represents the individual's facial features. To ensure that the images are properly pre-processed, it is necessary to use landmarks from the individual's face. Typically, these landmarks consist of the eye, nose, and mouth points, which are used to ensure that the image is aligned correctly and scaled.

We tested a single instance of a state-of-the-art face recognition model for our pipeline. This decision was based on two factors: the model's negligible inference-time compared to face detection and its near-perfect accuracy. Therefore, our primary focus was not on selecting the best-performing face recognition model but optimizing the pipeline's overall efficiency.

Arcface

Arcface [51] is a SOTA face recognition method that uses a neural network-based approach to extract discriminative features from faces. The technical details of Arcface include a modified ResNet architecture with a large embedding size, a novel angular softmax loss function, and specific optimization techniques. The ResNet architecture consists of several convolutional layers, which extract features from the input face image. The embedding size of Arcface is a 512-dimensional floating point array.

Arcface is trained using a custom loss function (*Arcface loss*), based on cosine similarity between features because it enforces more inter-class discrepancy. Different distance functions are used for comparing two embeddings in practice. Typically, the L2 loss function is used as distance measurement. However, in certain applications different distance functions are preferable. For example, a zero knowledge proof might need an inner product for efficient calculation, therefore cosine distance might be the preferred function.

6.2 State-of-the-art face recognition pipeline

The typical SOTA setup for image-based face recognition consists of the following components:

$$\text{Camera} \rightarrow \text{Detection} \rightarrow \text{Recognition} \rightarrow \text{Comparison}$$

The size of the retrieved camera image heavily influences the inference time. We assume that the camera produces 4k images. In alignment with Section 3.1.3, for the default pipeline, we use Retinaface [50] as face detection model and Arcface [51] as face recognition model.

Two embeddings are compared using a distance function. There has been no study on the impact of using different distance functions during inference. Therefore, we evaluated the impact of three popular distance metrics used with Arcface, namely absolute, L2, and cosine distance. We calculated the embeddings of the 6,000 test image-pairs from the LFW dataset and followed their protocol to verify the accuracy of Arcface using different distance metrics.

The precision-recall plot presented in Fig. 6.1 indicates only minor differences, which are only visible if we zoom in on the plot. The inference time is not affected significantly either; our benchmark indicates roughly 1 µs computation time for all three variants (L2: $1.0939\mu s \pm 3.9ns$, Cos-Dist: $1.1549\mu s \pm 20.9ns$, absolute: $1.0956\mu s \pm 8.6ns$)[1].

Our findings reveal that the choice of distance metric does not have a significant effect on the analysis outcome. Thus, due to popular use, the L2 norm is used for the rest of this thesis.

Decision

[1] Timing information has been measured with criterion (https://docs.rs/criterion)

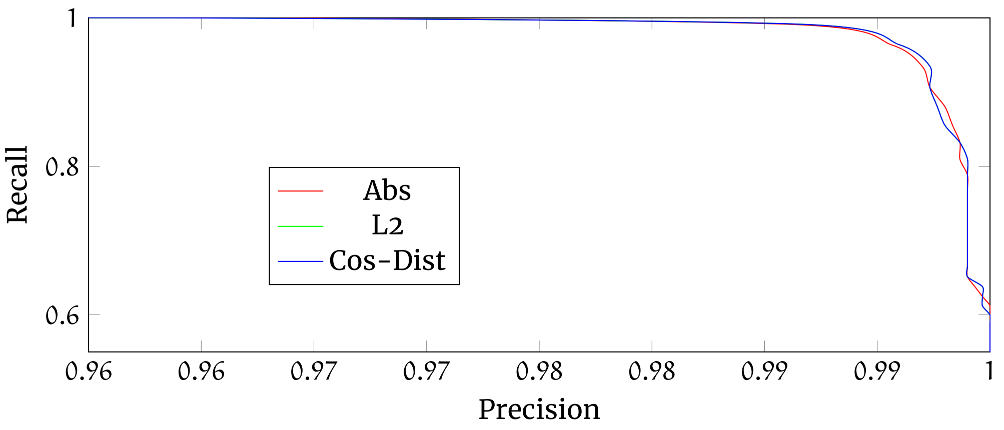


Figure 6.1: Different distance functions for Arcface. Notice the magnified scale; plotting the whole spectrum (0-1) would yield no discernible distinction. The green line is not visible, as using L2 and COS distance functions yields an identical precision-recall curve. The Area Under Curve (AUC) is not significantly different either: $AUC_{L2} = 0.99884653$, $AUC_{ABS} = 0.9988512$, $AUC_{COS} = 0.99884653$.

This gives us the following architecture for our default pipeline:

$$\underbrace{\text{Camera}}_{\text{4k images}} \rightarrow \underbrace{\text{Detection}}_{\text{Retinaface}} \rightarrow \underbrace{\text{Recognition}}_{\text{Arcface}} \rightarrow \underbrace{\text{Comparison}}_{\text{L2 Norm}}$$

6.2.1 Performance baseline

In order to establish a baseline for the performance, we implemented the pipeline in Rust using Tensorflow Lite (Retinaface and ULFGFD) and OpenCV (YuNet and Haarcascade). All benchmarks are executed on a Jetson Nano[2], with an NVIDIA Maxwell GPU and a Quad-core ARM Cortex-A57 MPCore CPU.

There are two distinct performance metrics:

1. With respect to **time**: We established benchmarks using the Rust performance measurement framework Criterion [76]. To ensure statistical significance and reliability, each component underwent 100 iterations, and the reported time is based on the median of these runs. The variance is less than 4.8 % of the value for all components. It is noteworthy that the times reported are calculated per image, with Retinaface requiring a total of 91 seconds for inference.

$$\underbrace{\text{Camera (4k)}}_{0.02\text{s}} \rightarrow \underbrace{\text{Retinaface}}_{91\text{s}} \rightarrow \underbrace{\text{Arcface}}_{0.071\text{s}} \rightarrow \underbrace{\text{Comparison}}_{0.000028\text{s}}$$

Retrieving the 4k image from the camera is possible at that frequency because hardware acceleration and MJPG compression are used.

[2] https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-nano/

2. With respect to **accuracy**: We use the 6,000 face comparisons proposed by LFW [159] and run the face recognition pipeline on it. If multiple faces are found, the one closest to the center is used, as the LFW images are preprocessed in that way. Retinaface manages to find all faces. As LFW primarily features single-person portraits, this accuracy was expected. Arcface uses the best threshold on that dataset to decide if the two faces are from the same person.

$$\text{Camera (4k)} \rightarrow \underbrace{\text{Retinaface}}_{100\,\%} \rightarrow \underbrace{\text{Arcface}}_{99.3\,\%} \rightarrow \text{Comparison}$$

6.2.2 Baseline improvements

Time-performance (1.5 minutes per 4k image) is arguably too slow for real-time performance. Most time (99.9 %) is spent on Retinaface. There are two options to reduce the inference time:

1. Reduce the input dimension, which yields the following time-performance:
   - 4k (3840x2160px): 91.24 s
   - Full HD (1920x1080px): 11.52 s
   - HD (1280x720px): 5.13 s
   - SD (640x480px): 1.72 s

   How is accuracy-performance affected if input dimension is reduced? The theoretical lower limit is detecting people of size 16 px x 16 px, as this is the smallest anchor used by Retinaface. We tested if such small faces are detected in practice. Starting with LFW's image size of 250 px x 250 px, we run our face recognition pipeline over all (test) images to determine the detected face size. Subsequently, the images were scaled down by 50 pixels, and the experiment was repeated until the image size was 50 x 50 px. The resulting face sizes were recorded. Fig. 6.2 illustrates the widths and heights in pixels for detected faces. It is apparent that the smallest anchors are not only used for sub-features (for use in higher levels of the FPN [119]), but also to detect faces directly. Interestingly, the smallest detected face has a dimension of 10 px x 14 px. This is smaller than the smallest anchor (16 px x 16 px) and is possible because the network refines its predicted bounding box in later stages.

   Despite the successful detection of faces, there is no guarantee that the image has enough information for face recognition to recognize a person. Therefore, we created another experiment by performing the same shrinking of the images as before. An embedding of the scaled down version of the image is (L2) compared to the embedding of the full image. The results are plotted in Fig. 6.3.

   As anticipated, our analysis reveals a distinct threshold at approximately 40 x 30 pixels, beyond which facial recognition accuracy is substantially diminished.

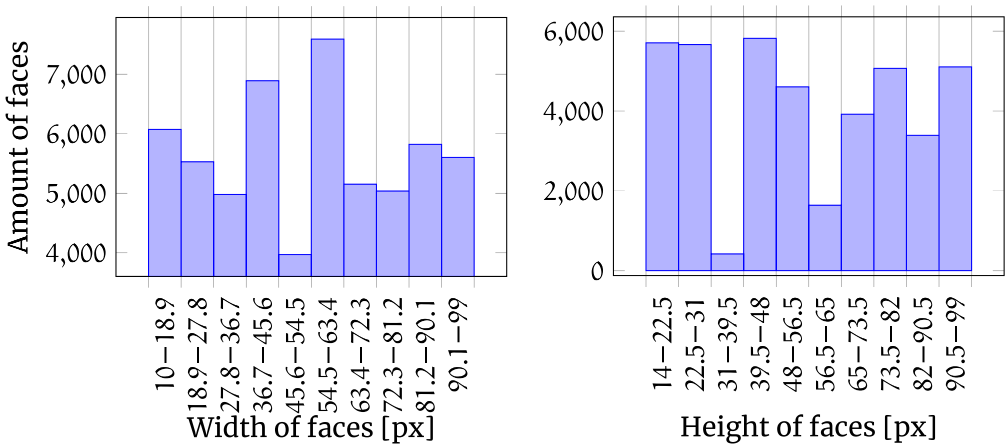


Figure 6.2: The sizes of detected faces using Retinaface. Sizes larger than 99 pixels are not displayed as our focus was on identifying the smallest detectable faces.

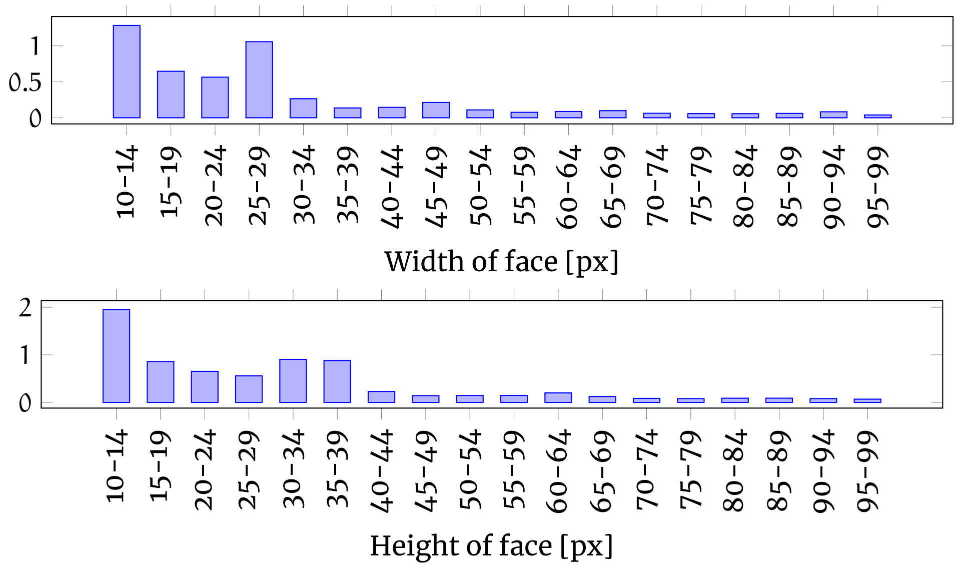


Figure 6.3: L2 distance to reference embedding (full size face) using different face sizes (smaller is better).

Even though an image of SD quality still has an inference time of 1.7 seconds, it skips 99.69 % of potential input data (25,600 vs 8,294,400 pixels).

We can calculate the real-world impact of this dimension reduction. For this calculation, we need a few hardware assumptions.

- **Face dimensions** Being able to detect an object depends on its size. Since we want to detect a face, we have to assume the dimensions. The US Department of Defense measured the width (bitragion breadth) and height (menton-crinion length) of the face to be between 12–15 cm and 15–21 cm, respectively [54]. We want to find the lower limit of face recognition pipelines possibilities. Therefore, we use the upper end of the face dimension scale:

$$\text{face}_{\text{width}} = 0.15\text{ m}, \text{face}_{\text{height}} = 0.21\text{ m}$$

- **Camera** For the camera, we assume a typical 70 mm focal length with a full frame 35 mm sensor:

$$\text{camera}_{\text{focallength}} = 0.07\text{ m}$$

$$\text{camera}_{\text{imagewidth}} = 35\text{ mm}, \text{camera}_{\text{imageheight}} = 24\text{ mm}$$

Figure 6.3 demonstrates that facial recognition can reliably commence at sizes as small as 40 x 30 pixels.

$$\text{object}_{\text{width}} = 30\text{ px}, \text{object}_{\text{height}} = 40\text{ px}$$

We can now calculate the maximum distance in millimeters of a person with respect to the camera, such that the face is still recognizable:

$$\text{distance} = \frac{\text{camera}_{\text{focallength}} \times \text{pixel}_{\text{width/height}} \times \text{face}_{\text{width/height}}}{\text{object}_{\text{width/height}} \times \text{camera}_{\text{imagewidth/height}}}$$

If we use $\text{pixel}_{\mathtt{width/height}}$ of 640 and 480 respectively, we can detect faces up to 6.4m. With a $\text{pixel}_{\mathtt{width/height}}$ of 3840 and 2880 this distance increases to 38.4m.

2. Use a different, more lightweight model. Due to the use of a large backbone network (ResNet [75]) and its computationally heavy use of feature pyramid networks [119], the inference time of Retinaface is slow. There are lighter networks with fewer parameters, such as ULFGFD. One major deficiency of fast face detection algorithms is their tendency to produce false positives. Retinaface, on the other hand, has been shown to have a very low false positive rate, making it a more reliable option for these types of applications. Fig. 6.4 shows the self-reported confidence of the face of ULFGFD. The first bar at $x = 0$ represents the 9.2 % of the cases where ULFGFD does not detect a face. 77 % of the images have a confidence of over 90 %.

   As face recognition expects a pre-processed image and this pre-processing depends on the location of landmarks, it is not possible to calculate face recognition accuracy with ULFGFD.

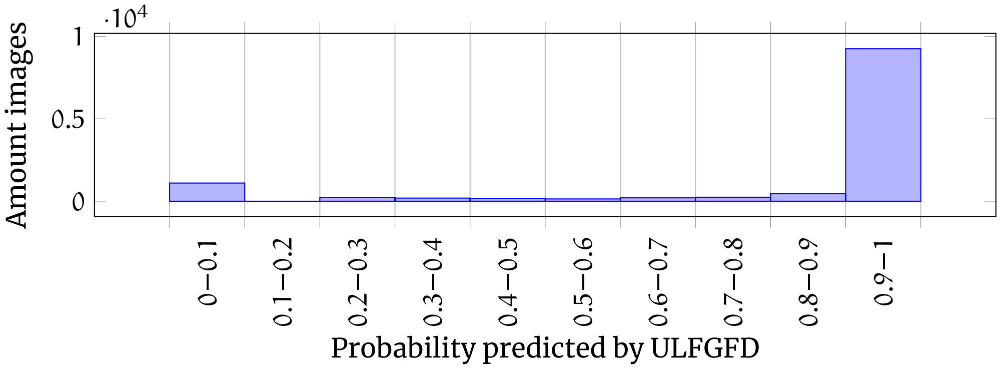


Figure 6.4: 77 % of images have more than 90 % probability.

6.3 Inference-time/accuracy tradeoff

The accuracy of face detection models has been extensively studied and reported in modern research (as demonstrated by the reported metrics described in Section 6.1.1). However, an often overlooked aspect in the evaluation of these models is their inference time. This information is important, as a slow inference time can lead to delays and long queues, compromising the effectiveness of the system (cf. Section 6.2.1). Inference time can also impact the scalability and cost-effectiveness of a face detection system, as a slow model may require more powerful hardware or computing resources to achieve the desired performance. Furthermore, inference time is especially important when considering the deployment of face detection models on embedded hardware. These devices often have limited computing resources and require models that can perform in real-time. Therefore, evaluating face detection models based on their inference time is essential for ensuring that they can be deployed effectively on embedded hardware and meet the performance requirements of real-world applications. Despite its importance in real-world deployment scenarios, none of the existing datasets currently available comprehensively address this aspect of performance evaluation. As a result, there is a significant gap in our understanding of the practical implications of face detection model performance in real-world settings.

This chapter evaluates SOTA face detection models with respect to these metrics. We assessed the performance and accuracy of four face detection models, Retinaface [50], ULFGFD [120], YuNet [62], and Haarcascade [217]. Figure 6.5 illustrates the space of performance-accuracy for current models. It is important to note that only the networks situated at the border of the performance-accuracy spectrum are relevant, and their selection depends on the specific application requirements. Different applications may require different points on the performance-accuracy spectrum, and our study provides insights into the trade-offs involved in selecting an appropriate face detection model for a given application. Our results, depicted in Fig. 6.5, clearly show that Haarcascade has a detection failure rate of over 50 %, even for the relatively simple portrait-like datasets such as the LFW dataset. Retinaface achieves high accuracy but requires high inference time, while ULFGFD has lower accuracy but a faster inference time.

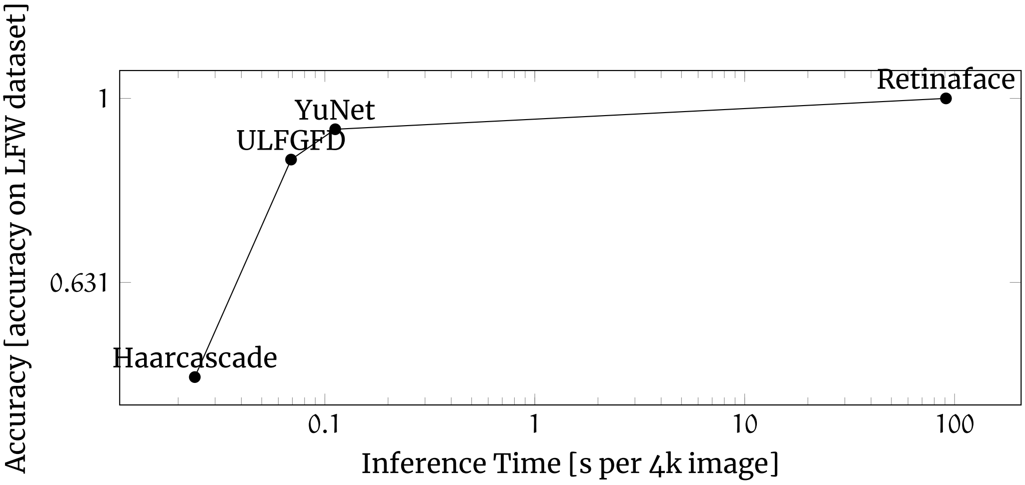


Figure 6.5: The figure illustrates the trade-off between inference-time and accuracy for various face detection networks. The x-axis represents the inference time, while the y-axis represents the accuracy of the networks. The solid line in the figure represents the Pareto frontier, which is the optimal trade-off between accuracy and inference time.

6.4 Fast and accurate face recognition pipeline

In order to optimize face detection for both speed and accuracy, we propose an approach that combines two algorithms with distinct characteristics in the inference/time spectrum to harvest the strengths of each. A fast algorithm is used as a proposal generator to create possible face detections quickly. We prioritize minimizing false negatives in the proposal generator, as the subsequent algorithm can verify false positives. While our analysis shows Haarcascade to be the fastest method, it misses more than half of the faces, even in the easy LFW dataset. Therefore, we use ULFGFD as our algorithm for generating proposals. These proposals are then confirmed and augmented with face landmarks by a more accurate algorithm, Retinaface. This yields the following pipeline:

$$\text{Camera (4k)} \rightarrow \text{ULFGFD} \rightarrow \text{Retinaface} \rightarrow \text{Arcface} \rightarrow \text{Comparison}$$

As discussed in Section 6.2, face recognition requires face dimensions of at least 30 px x 40 px. We recommend using face images with dimensions of 50 px x 65 px or larger to achieve higher accuracy. Our experimental results indicate that performance degrades when face images are smaller than this threshold.

As the pipeline should take bounding box errors of ULFGFD into account, we performed a systematic search on a grid of possible dimensions and performed additional benchmarks on Retinaface with respect to inference time:

- 150 px x 150 px: 0.169 s
- 125 px x 125 px: 0.093 s
- 100 px x 100 px: 0.052 s
- 75 px x 75 px: 0.038 s

- 50 px x 50 px: 0.018 s

Subsequently, we constructed the complete pipeline and evaluated the accuracy of each individual component as follows:

$$\text{Camera (4k)} \rightarrow \underbrace{\text{ULFGFD}_{\text{th=0.05}}}_{96.45\,\%} \rightarrow \begin{matrix} \underbrace{\text{Retinaface}_{\text{50x50}}}_{73.2\,\%} \\ \underbrace{\text{Retinaface}_{\text{75x75}}}_{86.1\,\%} \\ \underbrace{\text{Retinaface}_{\text{100x100}}}_{98.3\,\%} \\ \underbrace{\text{Retinaface}_{\text{125x125}}}_{98.9\,\%} \\ \underbrace{\text{Retinaface}_{\text{150x150}}}_{99.7\,\%} \end{matrix} \rightarrow \begin{matrix} \underbrace{\text{Arcface}}_{92.3\,\%} \\ \underbrace{\text{Arcface}}_{95.3\,\%} \\ \underbrace{\text{Arcface}}_{97.3\,\%} \\ \underbrace{\text{Arcface}}_{97.7\,\%} \\ \underbrace{\text{Arcface}}_{98.3\,\%} \end{matrix} \rightarrow \text{Comparison}$$

A size of 100 px x 100 px for the Retinaface input is a good tradeoff between time and accuracy performance. With this, the entire pipeline runs on $\sim$ 4.7 FPS on a Jetson Nano and achieves an overall accuracy of 92.3 % ($0.9645 \times 0.983 \times 0.973$) on the LFW dataset.

Next, we can calculate both the inference time and accuracy of the entire pipeline with these three combinations of networks and compare them to existing algorithms. Fig. 6.6 clearly demonstrates that by integrating multiple different networks, the trade-off border in the inference-time/accuracy spectrum is increased and a better balance between these two metrics is achieved. This indicates the effectiveness of our approach in improving face detection performance.

Notably, the selection of suboptimal parameters, as observed in Fast50 and Fast75, can lead to an unexpected outcome where the desired effect is inverted. Specifically, this may result in a decrease in accuracy despite a slower inference time. Therefore, it is crucial to carefully select appropriate parameters by utilizing techniques such as analyzing the inference-time/accuracy plot to ensure the desired performance outcome is achieved.

Summary

Real-time face recognition particularly for applications in decentralized systems without large GPU clusters comes with several challenges, including the trade-off between accuracy and inference-time on constrained hardware resources. Achieving higher accuracy is desirable, but it often comes at the cost of longer inference-time, which is particularly problematic for embedded devices with limited processing power.

We first conduct a comparative study on different face recognition distance functions to address this challenge and introduce an inference-time/accuracy plot. We propose, that future datasets and models should include inference time as a metric for performance evaluation. This will allow researchers to better understand the trade-offs between accuracy and efficiency in real-world

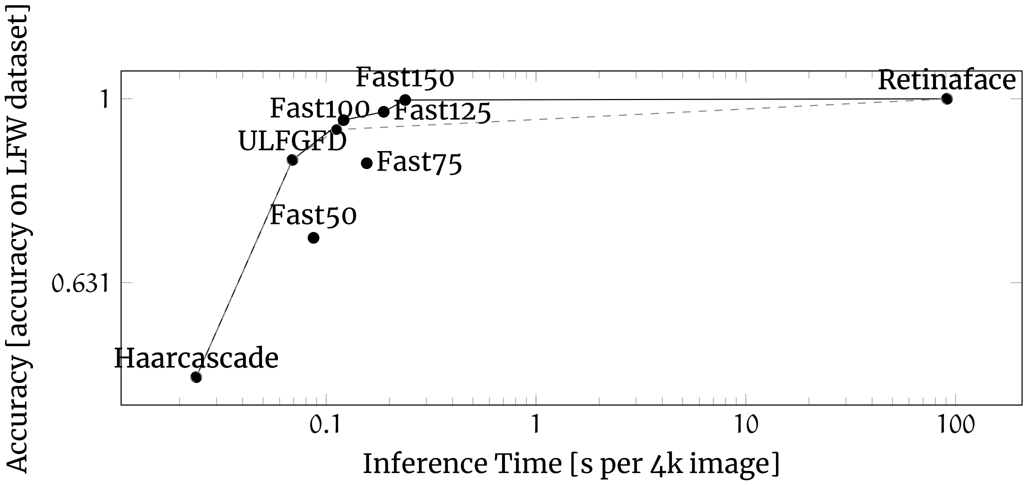


Figure 6.6: This plot adds our proposed models to the initial plot of Fig. 6.5, which is represented by the dashed line. As visualized in the solid line, our proposed Fast100, Fast 125, and Fast150 networks increase the Pareto front in the *inference-time/accuracy* spectrum Fast50 and Fast75 do not increase the border, as they are slower and have less accuracy than ULFGFD.

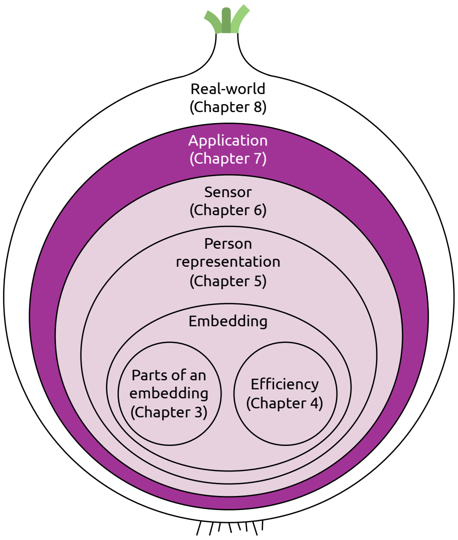
deployment scenarios, and enable the development of more effective and efficient face detection models that can be deployed in real-world applications. By including inference time as a metric, the practical relevance of face detection research is improved and models can be optimized for real-world deployment. We also introduced an inference-time/accuracy plot that enables the comparison of different face recognition models. Our analysis showed that different models have different strengths and weaknesses, and every application must strike a balance between inference-time and accuracy, depending on its focus.

To achieve optimal performance across the spectrum, we proposed a combination of multiple models with distinct characteristics. This approach allows the system to address the weaknesses of individual models and optimize performance based on the specific needs of the application. We demonstrated the practicality of our proposed approach by developing a multimodel face recognition pipeline. We utilized two face detection models positioned at either end of the inference-time/accuracy spectrum to achieve superior overall accuracy, reliability, and speed. Specifically, we employed the more accurate model only when necessary and the faster model for generating fast proposals, thereby improving the trade-off between inference-time and accuracy.

Overall, our proposed pipeline can serve as a guideline for developing real-time face recognition systems on embedded devices. By striking an optimal balance between the performance of different models, we can improve the overall accuracy, reliability, and speed of such systems and demonstrate this in Chapter 8.

Chapter 7

Biometric Domain Specific Sensor Language (BioDSSL)



Foundation

The foundation of this chapter is the following paper:
Hofer, Philipp, Philipp Schwarz, Michael Roland, and René Mayrhofer. 2024. BioDSSL: A Domain Specific Sensor Language for global, distributed, biometric identification systems. In *12th IEEE International Conference on Intelligent Systems (IEEE IS 2024)*. IEEE, Golden Sands, Bulgaria, (August 2024)

Building upon the theoretical advancements and practical implementations discussed in previous chapters, we transition from optimizing face detection for embedded systems in Chapter 6 to the development of an application framework. Chapter 6 highlighted the challenges and solutions for efficient real-time face recognition on resource-constrained hardware, establishing a foundation for practical deployment. Now, we introduce BioDSSL, a Domain Specific Sensor Language, which enhances the flexibility and efficiency of integrating multiple biometric modalities. This chapter bridges the gap between optimized facial feature processing and the dynamic management of various biometric sensors, enabling more robust and adaptable authentication systems.

As biometric identification systems become more ubiquitous, their complexity is increasing with the integration of additional sensors, aimed at minimizing error rates. The current paradigm for these systems involves hard-coded aggregation instructions, presenting challenges in system maintenance, scalability, and adaptability. These challenges become particularly prominent when deploying new sensors or adjusting security levels to respond to evolving threat models.

To address these concerns, this research introduces BioDSSL, a Domain Specific Sensor Language to simplify the integration and dynamic adjustment of security levels in biometric identification systems. Designed to address the increasing complexity due to diverse sensors and modalities, BioDSSL promotes system maintainability and resilience while ensuring a balance between usability and security for specific scenarios.

Furthermore, it facilitates decentralization of biometric identification systems, by improving interoperability and abstraction. Decentralization inherently disperses the concentration of sensitive biometric data across various nodes, which could indirectly enhance privacy protection and limit the potential damage from localized security breaches. Therefore, BioDSSL is not just a technical improvement, but a step towards decentralized, resilient, and more secure biometric identification systems. This approach holds the promise of indirectly improving privacy while enhancing the reliability and adaptability of these systems amidst evolving threat landscapes and technological advancements.

7.1 Complexity and rigidity of current systems

There are various types of biometric identification systems, leveraging diverse biological features. Some of the most common include fingerprint, facial, iris, and voice recognition. They are utilized for a wide range of applications, such as tracking students' attendance [90], opening doors [134], facilitating contactless payments for public transport tickets [30, 215], and even streamlining border control processes [113] (cf. Fig. 7.1).

As biometric traits are distinctive and unalterable, the potential misuse of such data raises significant privacy and security concerns. Furthermore, the complexity of integrating diverse sensors and modalities, along with the need for dynamic security levels, presents additional challenges in the development, deployment, and maintenance of these systems.

As biometric identification systems have become more pervasive, their complexity has escalated, primarily due to the integration of a variety of sensors [126, 163] and modalities [34, 185], all intended to minimize error rates and enhance system reliability. Each of these sensors and modalities comes with its own specifications, requirements, and compatibility issues, which increases the intricacy of these systems.

While hard-coded instructions were suitable for initial generations of biometric systems with a limited set of sensors and modalities, it is inflexible and challenging for more sophisticated systems. This rigidity is especially problematic

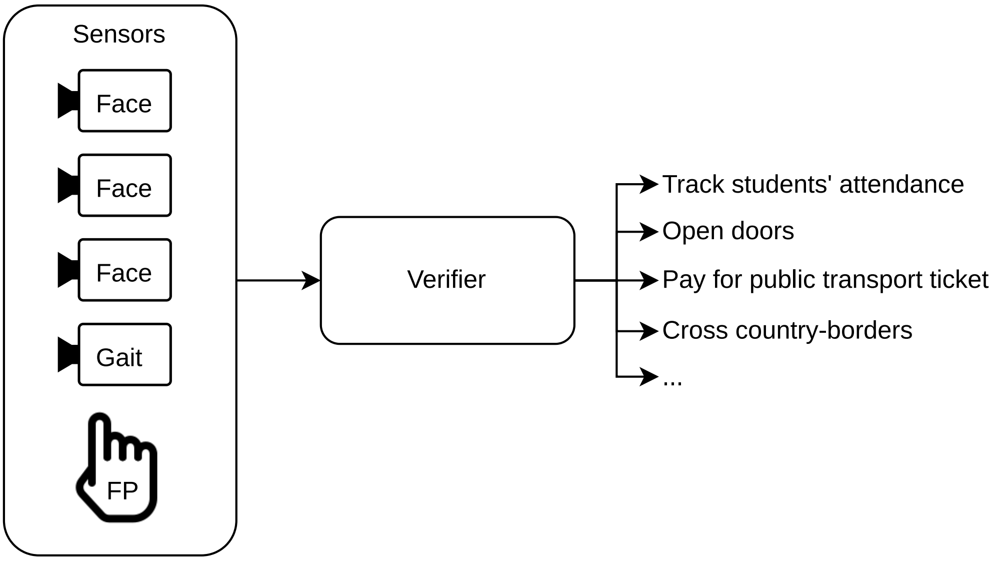


Figure 7.1: This figure shows the architecture overview of biometric systems, with examples of different sensors (*face*, *gait*, and *f*inger*p*rint-recognition). The sensors capture biometric data from people and send that representation (most commonly in form of a high-dimensional vector) to a verifier. The verifier receives this information from one or more sensors and can then decide to trust these sensings enough to perform an action.

when it comes to deploying new sensors or modifying system parameters to adapt to evolving threat landscapes or security requirements.

Further compounding the issue is the lack of a standardized, easy-to-use framework for integrating new sensors or adjusting system parameters. This lack makes it difficult for system developers and administrators to maintain, scale, and adapt their systems, leading to increased costs, longer deployment times, and potential vulnerabilities.

7.2 Proposed solution: BioDSSL

Given the growing complexity and rigidity of current biometric identification systems, there is a clear need for a more dynamic, adaptable, and scalable solution. To this end, we propose BioDSSL. One of the key advantages of BioDSSL is its ability to handle diverse sensor readings from various modalities. This allows for a more unified and efficient operation of biometric identification systems, regardless of the range of sensors and modalities they incorporate. By abstracting away the complexities of sensor integration and system configuration (Section 7.5.2), BioDSSL has the potential to reduce the time and effort required for system maintenance, while also improving scalability and adaptability in certain scenarios.

In the sections that follow, we focus on the concept and design principles of BioDSSL, explore its unique features and advantages, and discuss how it fosters decentralization in biometric identification systems.

7.3 Scope and goals

The primary objective of this chapter is to introduce BioDSSL and examine its potential role in enhancing the flexibility and security of biometric identification systems. The scope of our study includes an exploration of the design and features of BioDSSL, as well as an examination of how it addresses some of the current challenges faced by these systems.

We focus on the details of BioDSSL, discussing its concept, design principles, and approach towards decentralization of biometric identification systems. We further describe the unique features of BioDSSL as a tool in the biometric identification landscape, underlining its ability to simplify the integration of new sensors and dynamic adjustments of security levels (Section 7.5).

Furthermore, we focus on the practical implementation of BioDSSL (c.f. Section 7.6), while providing a detailed methodology, including steps for integrating new sensors and dynamically adjusting security levels using BioDSSL. This allows to balance usability and security as crucial elements for the efficient operation of biometric identification systems.

Moreover, we present case studies and experimental results demonstrating the efficacy of BioDSSL (Section 7.7). These real-world scenarios and experimental setups provide valuable insights into the practical application and advantages of BioDSSL. Additionally, quantitative and qualitative analyses of the results are provided to substantiate the improvements BioDSSL brings to biometric identification systems.

Lastly, we consider BioDSSL's impact on privacy and security (Section 7.8). Through this chapter, we aim to contribute to the ongoing dialogue about enhancing the flexibility, security, and efficiency of biometric identification systems.

7.4 Traditional approach

In this section, we focus on the historical context, evolution, and complexities surrounding the field of biometric identification systems (Section 7.4.1). We explore the role of diverse sensors and modalities in enhancing the robustness and accuracy of these systems (Section 7.4.2). We then address the challenges inherent in the present systems, detailing how their complexity and rigid structure makes system maintenance, scalability, and adaptability burdensome (Section 7.4.3). This section also reviews previous attempts at resolving these issues, drawing attention to their limitations and the gaps they leave unaddressed (Section 7.4.4). The collective understanding from this background study sets the stage for the introduction of our proposed solution to these challenges.

7.4.1 Evolution of biometric identification systems

Biometric identification systems have come a long way since their inception, evolving from basic systems with limited capabilities to sophisticated networks capable of handling multiple modalities and sensors. The earliest biometric systems were simple, employing single modality biometrics such as fingerprints or facial features for identification. As the technology advanced, these systems saw improvements in their speed, accuracy, and reliability. However, they remained largely static and rigid in their design, with fixed security levels and little flexibility to integrate new sensors or adjust to evolving threat landscapes.

In the last decade, the focus has shifted towards multi-modal biometric systems that integrate multiple biometric traits for more accurate and reliable identification [1, 34, 112, 179, 185, 226]. This shift has been driven by advances in sensor technology and computing power, along with the increasing need for robust and secure identification systems.

While these advancements have significantly enhanced the capabilities of biometric systems, they have also introduced new challenges. The integration of diverse sensors and modalities has made these systems more complex. Additionally, the increasing concentration of sensitive biometric data has raised privacy and security concerns.

7.4.2 Diverse sensors and modalities in biometrics

As discussed in the previous section, biometric identification systems have evolved to incorporate multiple sensors and modalities, enhancing their accuracy and reliability. This section focuses on the diversity of sensors and modalities currently employed in these systems.

Biometric sensors can be broadly classified into two categories: physiological and behavioral [45]. On the one hand, physiological sensors capture biometric traits such as fingerprints [100], face [162], iris [168], and palm prints [253], which are inherent to an individual and remain relatively stable over time. On the other hand, behavioral sensors capture traits such as voice [4], gait [218], and typing rhythm [5], which are unique to an individual but can vary based on factors like mood or health.

As for modalities, single-modal biometric systems use one sensor type to capture one biometric trait, while multi-modal systems use multiple sensor types to capture multiple biometric traits. Multi-modal systems offer several advantages over single-modal systems, including increased robustness to noise, greater resistance to spoofing, and improved identification accuracy [153].

However, the integration of diverse sensors and modalities in biometric systems is not without its challenges. Each sensor and modality has its own specific requirements and complexities, including different data formats, varying levels of sensitivity, and distinct comparison protocols. Furthermore, the dynamic nature of behavioral biometrics introduces additional layers of complexity, requiring systems to be adaptable and flexible.

Combining data from different modalities or sensors can be handled on different levels of fusion:

- Sensor-level fusion: This is the earliest stage at which fusion can occur. It involves integrating data from multiple sensors before any processing takes place. This approach can provide a rich dataset for identification but can also introduce significant complexity due to the need to manage raw data from diverse sensors.
- Feature-level fusion: Features are extracted from the sensor data, and the feature sets from different sensors are combined. This method has the potential for high identification accuracy because it uses detailed feature information. However, it requires a high degree of compatibility between feature sets, which can be challenging to achieve with diverse sensors and modalities.
- Score-level fusion: At this stage, each sensor or modality independently processes its data and outputs a score representing the confidence of a match. These scores are then combined to make a final decision. Score-level fusion is a popular choice because it offers a good balance between the amount of information used and the generalizability of the approach.
- Rank-level fusion: This method also involves independent processing by each sensor or modality, but instead of outputting scores, they output ranked lists of potential matches. These ranks are then combined to make a final decision. This method can be efficient and relatively simple to implement but may not utilize the available information as effectively as score-level fusion.
- Decision-level fusion: This is the final stage at which fusion can occur. Each sensor or modality independently processes its data and makes a yes/no identification decision. These decisions are then combined to make a final decision. This method is the simplest to implement but uses the least amount of information, potentially resulting in lower identification accuracy.

7.4.3 Challenges in current systems

With an understanding of the diversity and complexity of sensors and modalities in biometric identification systems, as well as the different fusion levels, we can now focus on the challenges that these systems currently face.

One challenge in current biometric systems is the integration of new sensors. As we have seen, each sensor and modality comes with its own specific requirements and complexities. Integrating a new sensor into an existing system can be a daunting task, often requiring substantial effort and modifications to the system.

Further, adjusting security levels in response to evolving threat models is another challenge. Given the static nature of many existing biometric systems, making such adjustments can be complex and time-consuming. The inability to quickly and dynamically adjust security levels can potentially leave systems vulnerable to emerging threats.

In addition, balancing usability and security presents a persistent challenge. On the one hand, systems must be secure and robust against spoofing attempts and noise. On the other hand, they must also be user-friendly, minimizing the time and effort required by users during identification. Striking the right balance is a delicate task that many current systems struggle with.

These challenges highlight the need for a solution that simplifies sensor integration, enables dynamic security adjustments and makes it easy to balance usability and security. In the following sections, we will see how BioDSSL is designed to address these challenges, improving the overall efficiency, adaptability, and resilience of biometric identification systems.

7.4.4 Previous attempts at solutions and their limitations

While there has been extensive research on various aspects of biometric systems, a structured way of specifying components in a biometric system has not been addressed in academic literature. Most studies have primarily focused on the intricacies of different fusion levels and detailed exploration of single modality systems (c.f. Chapter 2.1). Consequently, these investigations do not provide comprehensive solutions for integrating diverse sensors seamlessly into an existing system nor updating the pipeline.

7.5 BioDSSL: A Domain Specific Sensor Language

Given the challenges and limitations identified in current biometric identification systems, we propose a novel solution, BioDSSL. We describe its underlying concept, design principles (Section 7.5.1) and unique features (Section 7.5.2) to simplify and enhance the resilience of biometric identification systems.

7.5.1 Concept and Design Principles of BioDSSL

BioDSSL has been developed with the aim to alleviate challenges related to complexity and scalability inherent in biometric identification systems. It accommodates the fact that different verifiers or operators of biometric systems can have vastly different requirements based on the context and purpose of the system (cf. Fig. 7.2).

For instance, in high-security environments such as border control checkpoints, the verifiers may wish to rely on a single, highly trusted biometric device that has been extensively validated for accuracy and reliability. This might include sophisticated devices such as iris scanners or high-resolution fingerprint readers, and the associated high level of assurance is necessary given the potential risks involved. On the other hand, in less critical contexts where the primary focus might be convenience or throughput, the requirements for the

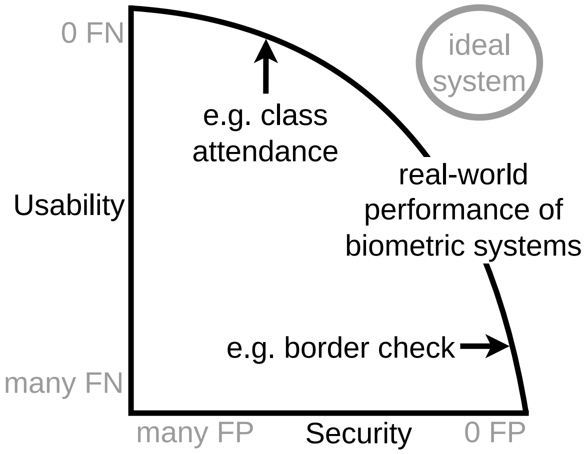


Figure 7.2: Different scenarios require a different trade-off between security and usability. In some cases (e.g. border control) false positives should be drastically reduced. In exchange, some false negatives might be acceptable, as additional (better) sensings could take care of these. On the other hand, in a different scenario (e.g. attendance tracking) the focus could be reducing false negatives, as the consequences are less severe than a false positive.

biometric system can be significantly less stringent. For example, in an educational institution tracking student attendance, less accurate but more expedient methods may be perfectly sufficient. In such a scenario, a simple facial recognition system or a fingerprint reader on a smartphone might be deemed adequate.

The fundamental concept behind BioDSSL is to provide a structured yet flexible language that can manage the configuration, integration, and operation of a wide variety of applications. Through this, BioDSSL aims to simplify the process of integrating new sensor modalities into existing systems, as well as provide mechanisms for dynamically adjusting the system's security settings as per situational requirements.

7.5.2 Unique Features and Advantages

Level of fusion

BioDSSL wants to use as much biometric data as possible for fusion without overly complicating the system or creating undue burdens when changes are implemented. To this end, BioDSSL adopts score-level fusion as a fundamental part of its design. This level of fusion is chosen because it retains a high level of information, but does not necessitate the re-training of complex models when changes, such as adding a new modality, are introduced. This is in contrast to sensor-level and feature-level fusion, which, while heavily researched in recent years [149, 165], typically require retraining of the network whenever changes are implemented. Given the dynamic nature of biometric systems, does not seem feasible to retrain networks each time a change is made. Score-level fusion, therefore, represents a more practical and efficient choice. It allows BioDSSL to accommodate changes in the system, such as the addition of new modalities or updates in security levels, without needing to undergo time-consuming and resource-intensive retraining processes.

Sensor tags

BioDSSL incorporates a tag system for sensors to increase flexibility and adaptability, allowing the system to respond effectively to a wide array of circumstances, whether anticipated or not. The tags can be as generic or as specific as required by the context. For our running examples, a tag could be *soft biometric* for a system tracking student attendance, or it could refer to a specific device model used at a border control checkpoint. By allowing the tags to be mutable and not fixed, BioDSSL can adapt to evolving circumstances and changing system requirements.

In more detail, useful tags for a sensor could include a universally unique identifier (UUID) for unambiguous identification, the modality (fingerprint, iris, face, etc.), the operator (who uses or manages the sensor), the modality class (soft or hard biometric), the certified by tag (indicating the certifying authority), and the location of the sensor. These tags enable a granular level of control and customizability.

7.6 Implementation

BioDSSL adopts a straightforward language syntax. The core elements of the language include tags (*TAG*), which are alphanumeric strings that can include hyphens, and values (*VALUE*), which can either be strings without quotes (*VALUE-NO-QUOTES*) or strings enclosed in quotes (*VALUE-WITH-QUOTES*). These elements are combined to create tag-value pairs (*TV*), which are semicolon-separated tag-value sequences (*TVS*). Each sensor has the mandatory *SECS* tag, denoting a floating-point value that defines the permissible duration, in seconds, for utilizing a reading. Additionally, it supports an arbitrary number of tag-value sequences using a comparison operator (“>” or “<”), along with a threshold level (*THRESHOLD*) associated with that sensor. The *THRESHOLD* represents a specific biometric parameter value that must be exceeded or not reached, depending on the operator, for the sensor's reading to be considered valid. A complete set of sensors (*AUTH*) is then defined as a comma-separated sequence of sensor definitions.

The language structure can be represented in augmented Backus-Naur form [39]:

```
TAG = ALPHA *(”-” / ALPHA) ;
VALUE-NO-QUOTES = 1*(ALPHA / DIGIT) ;
VALUE-WITH-QUOTES = DQUOTE 1*(VALUE-NO-QUOTES / SP) DQUOTE ;
VALUE = VALUE-WITH-QUOTES / VALUE-NO-QUOTES ;
TV = TAG ”=” VALUE ;
TVS = TV *(”;” TV) ;
```

```
THRESHOLD = FLOAT ;
SENSOR = ”SECS = ” INTEGER ”;” TVS (”<” / ”>”) THRESHOLD ;
JOIN = ”AND” / ”OR” ;
AUTH = [”(”] SENSOR *([”(”] JOIN SENSOR [”)”]) [”)”] ;
```

This language design, while simple, enables the representation of complex sensor configurations, allows dynamic adjustments of security levels and is able to accommodate a wide array of sensors and modalities.

The ability to adjust the confidence level and comparison operator directly addresses the need for dynamic security adjustments, catering to varying needs across different contexts. In the following section, we will explore how this implementation of BioDSSL is tested and validated through case studies and experimental results. This will provide a practical demonstration of BioDSSL’s efficacy in addressing the challenges in the current paradigm of biometric identification systems.

7.7 Case studies and experimental results

To evaluate the effectiveness and efficiency of BioDSSL, we conducted a series of experiments.

7.7.1 Experimental setup

Our experimental setup involved describing the application of BioDSSL in different biometric identification system contexts, ranging from high-security applications such as border control, to more routine scenarios like student attendance tracking. The chosen scenarios differed significantly in their security requirements, the diversity of sensors used, and the volume of biometric data handled. This diverse selection was intended to test the adaptability and versatility of BioDSSL.

We used a variety of modalities in our experiments, including both hard and soft biometrics. The tag system of BioDSSL allowed to manage this diversity and helped in the seamless integration of new sensors.

7.7.2 Case studies demonstrating the efficacy of BioDSSL

The case studies conducted show the flexibility of BioDSSL in accommodating a variety of system requirements and sensor configurations.

Student attendance tracking

The first case study focused on student attendance tracking for a lecture. In this setting, a single soft biometric could be sufficient to meet the system’s requirements, leading to this BioDSSL config:

```
AUTH = secs=60;operator=dept-a;
  modality=”soft biometric” < 1.0
```

This example demonstrates how BioDSSL can be used to manage a lower-security requirement setting, accommodating a soft biometric sensor and enabling easy adjustment of security settings based on the context. The tag system streamlined the integration of the soft biometric sensor. Moreover, the inherent adaptability of BioDSSL provides the department with flexibility for future expansions or changes. For instance, if the department decides to deploy a new sensor, even of a different modality, the system will continue to operate seamlessly, as long as the new sensor is also tagged as a *soft biometric* modality.

The adaptability of BioDSSL proves to be a valuable feature, providing flexibility for future expansions or changes. For instance, the department could decide to introduce a second authentication modality using a fingerprint scanner. By tagging the fingerprint scanner as a new modality, students who use the scanner would also be marked as present.

```
AUTH = secs=60;operator=dept-a;
  modality=”soft biometric” < 1.0 OR

  secs=300;operator=dept-a;
  modality=”fingerprint” < 0.04
```

An additional benefit of BioDSSL are decentralized deployments. If another department, physically located in the same hallway also want to use biometric attendance tracking, the same sensors can be used, provided that the sensor’s operator grants permission for this shared use. Without BioDSSL, each verifier would need to be individually configured and updated whenever there are changes in sensor usage or security protocols. This task becomes cumbersome and prone to errors with an increasing number of verifiers. However, with BioDSSL, changes can be implemented universally by merely updating the shared BioDSSL specification, significantly reducing the effort and potential for errors. For instance, if the department decides to introduce gait recognition sensors in multiple locations, the BioDSSL configuration can be effortlessly updated to accommodate the new modality, as shown below:

```
AUTH = secs=60;operator=dept-a;
  modality=”soft biometric” < 1.0 OR

  secs=300;operator=dept-a;
  modality=”fingerprint” < 0.04 OR

  secs=30;operator=dept-b;
  modality=”gait” < 1.0
```

Border control

On the other end of the spectrum, in the context of a high-security scenario such as border control, the system requirements differ significantly. The border control authority could rely on a specific, trusted device to ensure stringent security measures are met.

To integrate the trusted device into the BioDSSL system, the following configuration could be used:

```
AUTH =  secs=15;uuid=655f60a4 < 0.3
```

This implementation showcases the ability of BioDSSL to seamlessly incorporate specific, trusted devices within high-security applications. By specifying the device's unique identifier (*UUID*) and defining the appropriate security threshold, the system can effectively utilize the trusted device to enhance security measures at border control checkpoints.

However, in order to further enhance the border control system's capabilities, the integration of, for example, radar distance sensing can be considered. Radar distance sensing technology can provide valuable information about the physical proximity of individuals, which can be useful in identifying potential threats or unauthorized access attempts. To incorporate radar distance sensing into the existing BioDSSL system, the sensor configuration can be extended as follows:

```
AUTH =  secs=15;uuid=655f60a4 < 0.3 AND
   secs=15;modality=”radar distance” < 0.5
```

By including the additional modality of radar distance and assigning an appropriate threshold, the system can leverage radar distance sensing to complement the existing trusted device. This combination of sensors enables a multimodal approach to security, incorporating both the trusted device and radar distance sensing to enhance threat detection and to ensure a robust border control system.

The use of parentheses and the logical operators "and" and "or" (*JOIN*) in BioDSSL enables the creation of more complex scenarios and enhances the system's flexibility. By enclosing sensor configurations within parentheses, it becomes possible to group conditions and establish precedence when evaluating them. This allows for the specification of intricate requirements and the logical relationships between different sensor modalities or thresholds.

7.8 Attacks

While BioDSSL may enhance the flexibility of biometric identification systems, it is essential to assess its impact on privacy and security to ensure these benefits do not come at the expense of user protection. This includes understanding potential vulnerabilities and considering potential attack vectors.

One potential threat scenario involves a rogue sensor that could deceive the system by falsifying specific tags. In this scenario, an attacker could manipulate a sensor to replicate the tags associated with a trusted sensor, leading the system to accept fraudulent biometric data. This type of attack is similar to a cybersecurity technique known as "spoofing", where an unauthorized entity assumes the identity of a trusted entity to exploit the system's vulnerabilities.

To address this issue, several mitigations commonly employed against spoofing attacks can be applied in this instance as well. These include:

- Authenticating sensors: Implementing a robust authentication mechanism that verifies the identity and integrity of each sensor can prevent rogue sensors from infiltrating the system. This ensures that only trusted sensors are accepted, reducing the risk of fraudulent data.
- Digital signatures for tags: Utilizing digital signatures for the tags issued by trusted entities. These entities can range from a specific institution or organization to a more global or national authority. For instance, in a system designed for tracking student attendance, only those tags signed by the responsible educational institute could be trusted. Alternatively, certain applications may opt to trust tags signed by a more overarching issuer, such as a national regulatory body.

By implementing these mitigations, the system can reduce the risk of spoofing attacks and enhance its overall security posture in the face of rogue sensor threats.

The modular nature of BioDSSL allows for the integration of additional security measures as they become available or necessary. This could include cryptographic verification methods, dynamic tag assignment, or sophisticated anomaly detection algorithms to identify and isolate potential rogue sensors.

Summary

In conclusion, this chapter has presented BioDSSL as a solution to the escalating complexity of biometric identification systems. By addressing the challenges associated with system maintenance, scalability, and adaptability, BioDSSL offers a systematic and repeatable language for integrating new sensors and dynamically adjusting security levels based on specific use cases.

The decentralization of biometric identification systems is a key focus of privacy-conscious biometric identification. BioDSSL contributes to the dispersal of sensitive biometric data across various nodes, enhancing privacy protection and reducing the potential damage from localized security breaches. This step towards decentralized and resilient systems aligns with the progressive interconnectivity of our world.

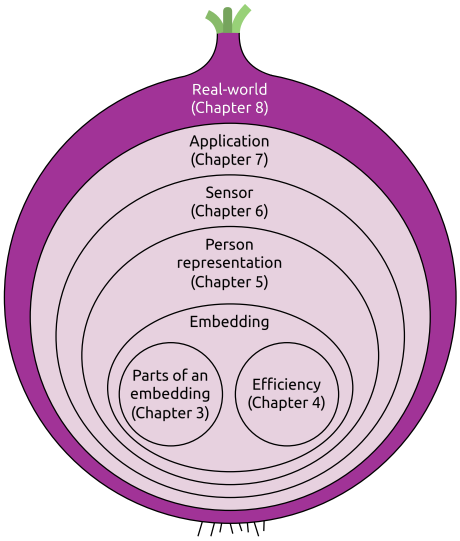
The adoption of BioDSSL not only improves technical aspects but also holds the promise of indirect benefits. It enables the efficient operation of biometric identification systems by handling diverse sensor readings from multiple modalities.

As biometric identification systems continue to become ubiquitous, BioDSSL offers an easy way for striking an optimal balance between usability and security.

Moreover, BioDSSL's adaptable design allows for future improvements and advancements, providing opportunities to enhance the overall architecture and further reinforce security measures. For example, in the future BioDSSL could be enhanced by offering encryption mechanisms, access control policies, and anonymization techniques to ensure the protection of biometric data during transmission or storage.

Chapter 8

When Theory Hits Reality: Living lab prototype and Digidow integration



As we approach the conclusion of this thesis, we shift our focus from the in-depth analyses presented in Chapters 3 through 7 to a broader examination of practical applications and potential synergies of our research findings. To demonstrate that the outcomes of this study extend beyond theoretical significance and can indeed enhance real-world applications when combined, we have developed a living lab prototype within the context of the CDL Digidow (c.f. Section 8.1).

This chapter begins by explaining the parts of Digidow that are relevant to this thesis, including its main components and how they interact (Section 8.1). Next, the focus shifts to the living lab prototype (Section 8.2), covering:

- **Legal assessments:** Conducting legal assessments is essential when dealing with biometric data to ensure compliance with data protection laws and regulations, such as obtaining necessary permits and performing Data Protection Impact Assessments (DPIAs).
- **Scenarios addressed:** The prototype addresses two main scenarios, demonstrating its versatility and practical relevance in real-world settings.
- **Hardware selection:** The choice of hardware is important for the system's performance and reliability. The selection process and the reasons behind choosing specific components are explained.

- **Programming language:** The decision to use the Rust programming language for implementation is due to its performance, safety features, and suitability for system-level programming.

A detailed description of the Sensor component follows (Section 8.3), highlighting the incorporation of improvements from previous chapters and showing how theoretical work translates into practical enhancements. The sensor is divided into different components, and reusable libraries (sensor-lib and face-lib) are created. This modular approach allows for easier extension and future work, removing the need for duplicate code:

1. **face-lib:** The face library implements Rust's state-of-the-art, off-the-shelve face recognition pipelines, including the combined face detection model described in Chapter 6.
2. **sensor-lib:** This library handles registrations, automatically manages their lifespan, and processes the sensing events.

Finally, the combination of these libraries to create an efficient pipeline capable of processing three frames per second using 4k cameras in two different settings (single-door and hallway scenarios) is demonstrated.

The chapter concludes by discussing real-world results, reflecting on how theoretical improvements are performed in practice and the insights gained from this implementation (Section 8.4).

8.1 Digidow

Digidow[1] is a digital identity system designed to address the growing need for secure and private identity verification in both digital and physical interactions. Recognizing the limitations and risks associated with centralized systems like Aadhaar, Digidow aims to offer a decentralized, user-centric alternative that enhances privacy and control over personal data.

8.1.1 Components

The primary focus of this thesis is the sensor, an essential component of biometric systems. However, the sensor alone does not constitute the entirety of such systems. At a high level, biometric systems typically require four components:

1. Sensor: Captures biometric data from individuals.
2. Identity Management System: Responsible for securely managing identities and storing personal biometric templates.
3. Identity Provider/Issuing Authority: Attests that a personal biometric template truly belongs to a specific individual, ensuring the authenticity of the stored biometric data.

[1] https://digidow.eu

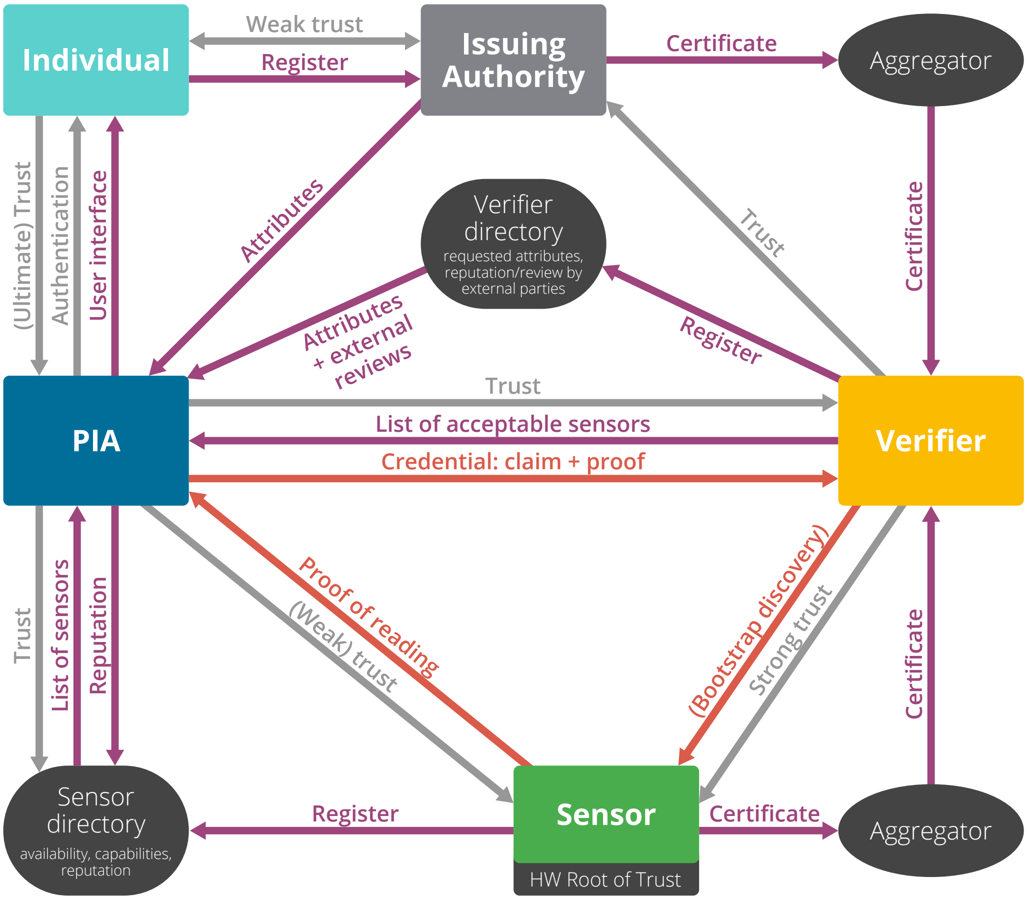


Figure 8.1: The components proposed by the Digidow project [133].

4. Verifier/Authorizer: Authorizes specific users to perform designated actions based on their authenticated biometric identity.

These components exist in both centralized and decentralized architectures, though their boundaries may be less distinct in centralized systems.

In Digidow, these components are implemented as follows (c.f. Fig. 8.1):

1. **Sensors** are used to link physical interactions with digital identities. These devices collect biometric data from individuals and authenticate it against the information stored in their PIAs. The system is designed to ensure that this biometric data is handled securely, preserving user privacy.

2. **Personal Identity Agents (PIAs)** securely store an individual's identity attributes and manages access to them. This approach allows users to retain control over their data, deciding where it is stored and who can access it. PIAs can be implemented on various platforms, including smartphones, home servers, and cloud services, providing flexibility and resilience.

3. **Issuing authorities** in the Digidow system validate and issue digital identity attributes. These authorities can range from government bodies to private organizations, broadening digital identities' applicability. The decentralized nature of these authorities helps distribute trust and reduces the risk associated with any single entity having too much control.

4. **Verifiers** are entities that need to authenticate individuals to grant access to services or resources. They interact with PIAs to obtain and verify the necessary attributes, ensuring secure and efficient access control. This setup is designed to handle a high volume of requests while maintaining data integrity and confidentiality.

8.1.2 Interaction

The sensor has two interactions with other Digidow components, specifically with the PIA.

- In Digidow, the PIA plays an active role: whenever it anticipates that the person it represents may want to perform an action, such as opening a door, it preemptively registers with the sensor. This registration essentially communicates:
  - "I might show up in the near future." → Implicit statement, transferred by the fact that the sensor receives this registration.
  - "You can identify me with that information." → This refers to a biometric identifier, typically an embedding for most modalities. Alternatively, it could be MPC contact information if the parties prefer not to share raw embeddings.
  - "Please contact me when this happens." → This is the callback onion address [209] of the PIA, which refers to a specific service hosted on the Tor network. An onion address is a unique identifier for a service within the Tor network. The use of onion addresses allows services to be accessed anonymously without revealing their physical location or IP address.

  Listing 8.1.2 shows an example of a registration.

```
{
   "@context":[
       "https://www.w3.org/2018/credentials/v1",
       "https://digidow.eu/v1",
   ],
   "credentialSubject":{
      "callback-url":"http://
    fbdcwqqjt4ainuwmwxbh7ns6vhqixser2piq4l6lhwifarespz447pyd.onion/",
      "embedding":[
        // embedding removed for brevity
      ]
   },
   "expirationDate":"2024-08-15T11:53:33UTC",
   "issuanceDate":"2024-08-15T10:53:33UTC",
   "issuer":"JKU",
   "proof":{
      "signatures":{
         "embedding":"odH9BZAOXBXgLPrAyQ45plZMi9dv/LXWLPlgcFmNngkcqzY+81
    KNN2wu1Ykjo+
    V6Y1ErdsLBaeXoQQNwraEf6MUFFJuP3VH17L5kJP8oAcZyIGB6OUM3P2jcQfsn7IAwVH/
    fmLtJRxjMZTzu8WmXzA=="
```

```
      },
      "type":"BBS+"
   },
   "type":"registration"
}
```

- If the sensor detects a registered person, it initiates a sensor push containing the following information:
  - Meta-data...
    - ... indicating that this is part of a Digidow transaction

      ```
      [
      "https://www.w3.org/2018/credentials/v1",
      "https://digidow.eu/v1"
      ]
      ```

    - ... specifying who issued this data

      ```
      "Self"
      ```

    - ... timestamp of the creation

      ```
      "2024-09-24T12:34:57UTC"
      ```

  - The specific action the sensor believes the user intends to perform (e.g., “door-opening-data”)...

    ```
    "door-opening-data"
    ```

  - ...and relevant details needed for the action:

    ```
    {
    "door-id": 42,
    "datetime": "2024-09-24T12:34:56UTC",
    "proposed-verifier": "http://ykozugfdo2urn6d3tvhge4eoaesftfe3rhefgagmrj4fumwphvajh3qd.onion/v1/push",
    "identity": // the registered embedding
    }
    ```

8.2 Living lab prototype

This prototype aims to validate the effectiveness and applicability of our research findings by integrating them into a functional biometric authentication system. By deploying this system in two distinct scenarios—a hallway and a single door—we can evaluate its performance, versatility, and reliability. The living lab environment serves as a testbed for our proposed ideas and provides insights into potential challenges and areas for further improvement.

8.2.1 Hardware

The hardware selection for the Digidow prototype was driven by the need to balance performance, cost, and practicality. The primary goal was to demonstrate that high-performance biometric authentication can be achieved using minimal and cost-effective hardware resources, which also prioritizes energy efficiency. This approach aims to prove that robust biometric systems can be implemented without relying on expensive, power-hungry hardware, making them more accessible and sustainable for widespread deployment.

- **Jetson Nano:** The core of the sensor system is the NVIDIA Jetson Nano, a compact computing platform specifically designed for AI and machine learning applications. The Jetson Nano was chosen for several reasons:

    - **GPU Acceleration:** It features a 128-core Maxwell GPU, providing computational power for parallel processing tasks such as image recognition and data processing. This GPU acceleration is crucial for real-time biometric authentication, enabling the system to process high-resolution images quickly and accurately.

    - **Power Efficiency:** Despite its computational capabilities, the Jetson Nano is designed to operate efficiently with low power consumption. To quantify its energy consumption, we employed a Power over Ethernet (PoE) switch for precise measurements. Our analysis revealed an average power draw of 10.82 watts if the full pipeline is continuously executed. This makes it suitable for deployment in various environments, including those where power resources are limited.

    - **Cost-Effectiveness:** Compared to other AI computing platforms, the Jetson Nano offers balance of performance and cost, making it accessible for experimental and practical applications without requiring significant investment.

  That said, the prototype's architecture is designed with modularity and flexibility in mind, allowing for easy hardware substitution. This design choice enables the system to adapt to various computational needs and resource constraints. For instance, the current hardware setup could be replaced with alternative configurations, such as a Raspberry Pi equipped with a Google Coral GPU accelerator[2] or the Raspberry Pi AI hat[3]. This adaptability ensures that the system can be optimized for different deployment scenarios.

- **Cameras:** For capturing biometric data, the prototype employs different camera setups tailored to specific scenarios:

    - **4k USB Cameras:** In the hallway scenario, 4k USB cameras are used to ensure high-resolution image capture. The higher resolution allows for more detailed facial recognition, improving the accuracy and reliability of the biometric system. These cameras are connected via USB, simplifying the setup and ensuring compatibility with the Jetson Nano.

[2] https://coral.ai/products/accelerator/

[3] https://www.raspberrypi.com/news/raspberry-pi-ai-kit-available-now-at-70/

- **1280x800 RTSP Camera:** In the single door scenario, a 1280x800 RTSP outdoor camera is utilized. This camera provides sufficient resolution for accurate facial recognition while being more affordable and outdoor proved housing into various environments. The RTSP protocol supports streaming, enabling the system to effectively capture and process real-time video feeds as an example for arbitrary RTSP cameras.

8.2.2 Programming language

Rust was chosen for the implementation of the Digidow prototype for several reasons, centered around performance, safety, concurrency, and modern language features.

- **Memory safety and security:** A standout feature of Rust is its strong emphasis on memory safety without needing a garbage collector. Rust's ownership model enforces strict rules on how memory is managed, effectively eliminating common bugs such as null pointer dereferencing and buffer overflows. These safety guarantees are vital for biometric systems, where security vulnerabilities can lead to severe breaches of sensitive data. By preventing such errors at compile-time, Rust enhances the security and stability of the Digidow system, ensuring that biometric data is handled securely and reliably.
- **Performance and efficiency:** Rust provides performance comparable to low-level languages like C and C++, which is critical for handling real-time biometric data processing. The language's zero-cost abstractions allow developers to write high-level code without incurring runtime overhead. This performance is essential for applications requiring quick and reliable biometric recognition, where delays or inefficiencies can significantly impact user experience and system reliability.
- **Concurrency:** Rust's design includes built-in support for safe concurrency, allowing the development of multithreaded applications that can efficiently utilize modern multicore processors. This is particularly beneficial for the Digidow system, which may need to process multiple biometric data streams simultaneously. Rust's concurrency model helps developers write concurrent code that is free from data races and other concurrency-related bugs, ensuring that the system remains performant and responsive under concurrent workloads.

8.2.3 Hallway scenario

Foundation

This subsection is primarily based on the content I developed for our website (accessible at https://www.digidow.eu/experiments/face-recognition-on-campus/). **Hofer, Philipp**, Philipp Schwarz, Michael Roland, and René Mayrhofer. 2024. Shrinking embeddings, not accuracy: Performance-Preserving Reduction of Facial Embeddings for Complex Face Verification Computations. In *14th International Conference on Pattern Recognition Systems (ICPRS 2024)*. IEEE, London, UK, (July 2024)

The goal of this project is to unlock doors without manual interaction if an authorized person intending to open a door is detected. This might be of particular interest when both hands happen to be full or when points of contact should be limited as much as possible, for example, when entering an operating room. Therefore, multiple non-moving cameras are placed around the second floor of JKU Science Park 3 in front of the Institute of Networks and Security, c.f. Fig. 8.2. The exact positions are marked in Fig. 8.3.

These cameras record the corridor, doors, and persons walking along the corridor. Camera sensor devices perform face detection in real time and initially discover any person in the image. The sensor verifies (again in real-time) if the person is a participant in this study. Additionally, the person's intent is predicted only to open a door if that is assumed to be the person's desire.

If a non-participating person is detected, any data about that person (i.e., the images and the fact that some unknown person was detected) is immediately discarded, and no information about the detection of such a person is stored. For participants (i.e., persons who explicitly opted in to the study), only the fact that a person has been recognized and the person's intent are stored and used to determine if a person is authorized to unlock the specific door.

The Digidow architecture consists of three components: The sensor and the personal agent agree upon whether the individual (person whose digital identity is managed by the personal agent) is in front of a door without revealing the sensor's raw images. The information that this particular person is standing in front of a particular door and has the intention to enter the room is authenticated by the sensor and provided to the personal agent. The personal agent then uses this authenticated proof to request the verifier to unlock the door given that the person is authorized to access the room.

If a person does not want to participate in this system, that person can still rely on the existing key card-based access control mechanism to access rooms. In that case, our experimental setup will not recognize the person and will, therefore, not store any data for these cases.

Legal aspects

Before diving into the technical aspects, since we want to process highly personal data, it is crucial to address the regulatory and legal considerations for the experiment setup. The Austrian Data Protection Authority (DPA) is Austria's

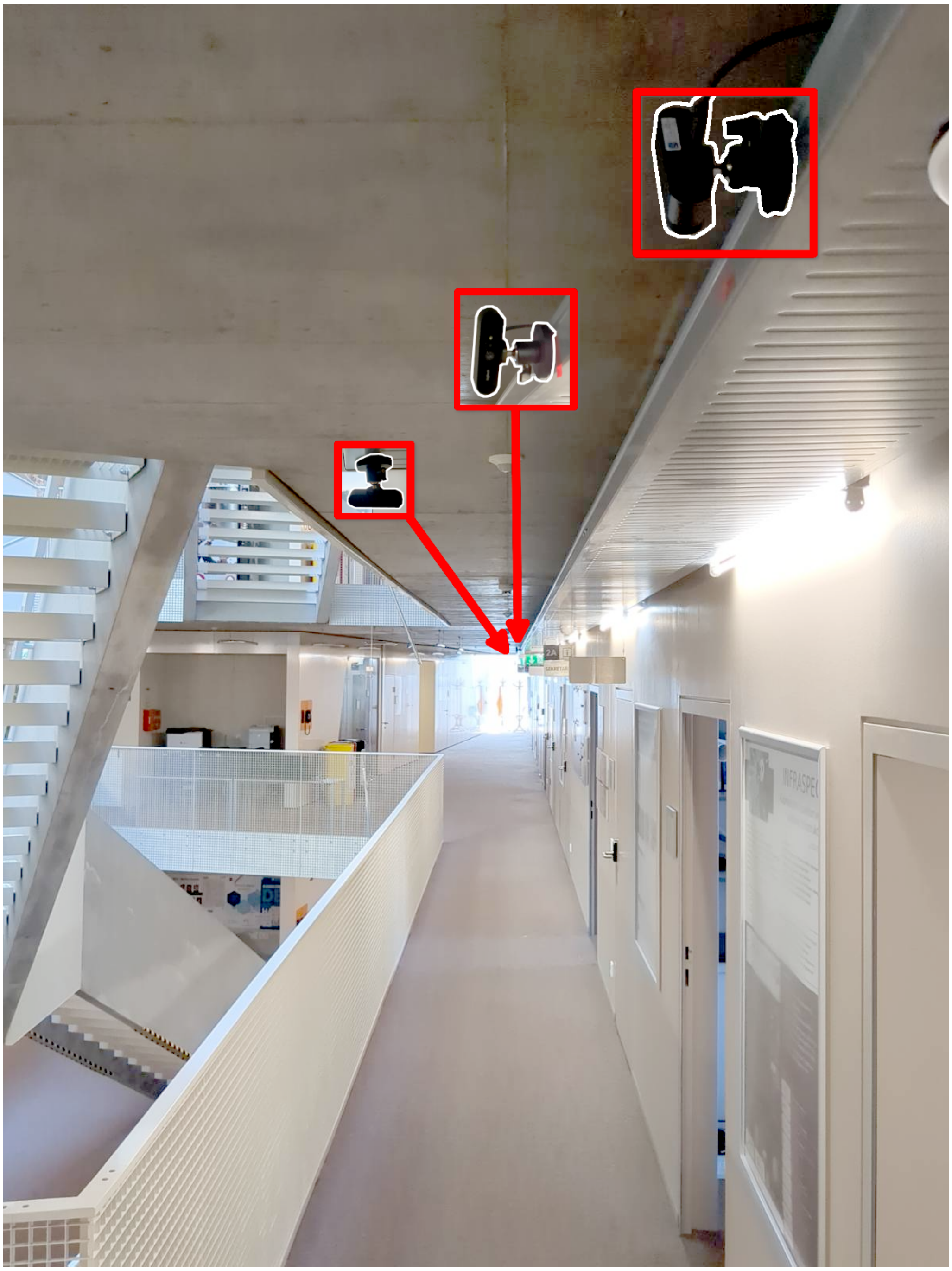


Figure 8.2: Physical setup of the hallway experiment. The corridor is monitored by three strategically placed 4k cameras, ensuring comprehensive coverage of the entire space. This arrangement allows for high-resolution observation and data collection across the length of the hallway.

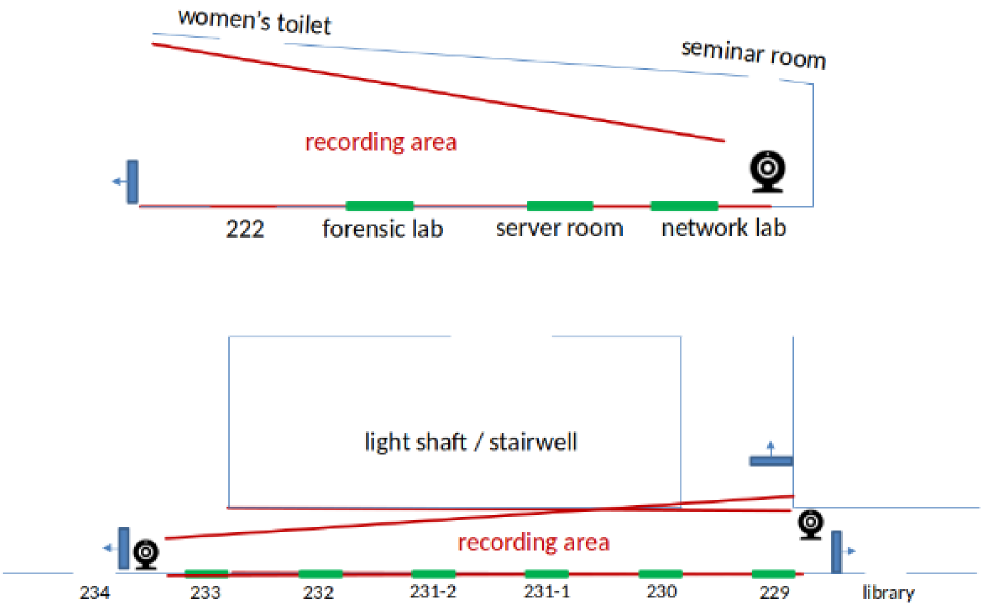


Figure 8.3: Top-down view of the experimental corridor showing the positions of three cameras strategically placed to provide full coverage of the hallway. Blue rectangles indicate the locations of signs informing individuals about the ongoing experiment and the use of facial recognition technology.

national supervisory authority for data protection. Legally, the space within JKU Science Park 3 is considered a public area during the day (though it becomes restricted at night when the doors are closed). Thus, since the experiment involves a semi-public space within the JKU Science Park 3, obtaining a permit from the DPA was necessary. We requested[4] and subsequently received the required approval[5].

It was not clear if a Data protection impact assessment (DPIA) is required for our experiment. Based on the guidelines of the Article 29 Data Protection Working Party (WP 248), a DPIA is necessary if at least two out of nine criteria[6] for likely high-risk data processing operations are met. Even if only to a small extent, our experiment fulfills three of these criteria:

1. *Systematic monitoring* is fulfilled because a larger area is continuously monitored.
2. *Sensitive data or data of a highly personal nature* is fulfilled because images are analyzed according to biometric characteristics.
3. *Innovative use or applying new technological or organizational solutions* is fulfilled because the automatic unlocking is a new application and also includes organizational aspects.

[4] Accessible at https://www.digidow.eu/experiments/face-recognition-on-campus/JKU-face-recognition-DSB-request.pdf
[5] Accessible at https://www.digidow.eu/experiments/face-recognition-on-campus/JKU-face-recognition-DSB-decision.pdf
[6] Accessible at https://ec.europa.eu/newsroom/article29/items/611236

Further details are outlined in our data protection impact assessment, accessible at https://www.digidow.eu/experiments/face-recognition-on-campus/JKU-face-recognition-DSFA-automatische-Tuerentsperrung.pdf.

8.2.4 Single door scenario

The single door scenario tests the system's performance and reliability in a different environment and setup compared to the hallway scenario. Here, a single camera is mounted directly above a door, focusing on capturing and processing biometric data for individuals approaching this specific entry point. This scenario is designed to evaluate the system's capability to operate effectively in a localized area where quick and accurate authentication is crucial.

The single-door setup involves a 1280x800 RTSP camera, chosen for its balance between resolution and practical deployment considerations, considering older in-the-wild cameras. This camera provides adequate detail for facial recognition while maintaining affordability and ease of integration. The camera streams real-time video to the Jetson Nano, where the biometric data is processed locally.

In this scenario, the system's primary objective is to adapt to different environments: Testing the system in a single door setting allows us to assess its adaptability to various architectural layouts and environmental conditions. Additionally, this single-door scenario enables the comparison of outdoor vs. indoor cameras, allowing us to test how different environmental factors, such as lighting and weather conditions, impact the system's performance. This scenario helps to validate the system's decentralized architecture. By deploying the system in a distinct location, we can demonstrate its ability to function independently and securely without relying on a centralized server. This decentralization enhances the robustness and scalability of the biometric authentication system, making it suitable for a wide range of applications, from small office spaces to larger, more complex facilities.

8.3 Sensor

Source code

The code is publicly available at https://git.ins.jku.at/proj/digidow/sensor and in Chapter A.

The sensor implementation for the Digidow prototype utilizes a modular approach, incorporating two core libraries to ensure flexibility, reusability, and efficiency: face-lib and sensor-lib (c.f. Fig. 8.4). This modular design is motivated by the need to create a robust and adaptable system that can be easily extended and maintained. By separating the functionality into distinct libraries, we can focus on specific tasks such as face detection, face recognition, and secure communication, thereby enhancing the overall scalability of the system. This approach also facilitates future improvements and integrations, allowing the system to evolve with advancements in biometric technology and security practices.

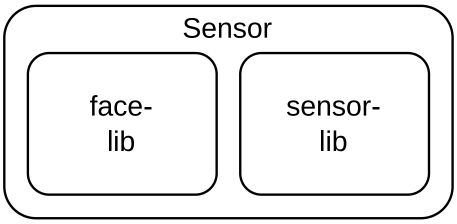


Figure 8.4: Components of the sensor code.

8.3.1 face-lib

Source code

This library is publicly available at https://git.ins.jku.at/proj/digidow/sensor-lib and in Chapter A.

face-lib is a Rust-based library that provides comprehensive functionalities for facial biometric processing. It supports both face detection and face recognition, employing state-of-the-art models to ensure high accuracy and performance.

Face detection The library provides functionalities to perform face detection on images and extract the embeddings from each face (c.f. Listing 8.3.1). The models used throughout this thesis, Retinaface and the faster ULFGFD, are currently supported. Additionally, it incorporates a combined face detection model described in chapter 6, enhancing detection reliability under diverse conditions. To illustrate the library's usage, here's a code example that demonstrates how to initialize the face detection pipeline, process an image, and retrieve the results. The library's flexible design allows users to easily switch between different detection models or utilize the combined approach based on their specific requirements for accuracy and speed.

```
use face::Detection;
use face::detection::{FaceProposal, FaceLandmarkProposal, FacialArea, Landmarks,
    Point, realface::RealFace};

let mut f = RealFace::new(0.7,
            "./static/detection/fastdet_640.onnx",
            "./static/detection/retinaface-150x150.tflite",
            "./static/detection/retinaface_anchors-150x150.json").unwrap();
let img = face::img_read!("static/test-images/person.png");

let result = f.inference(&img).unwrap();

assert_eq!(result, vec![
    FaceLandmarkProposal {
        face: FaceProposal {
            probability: 0.9986744,
            facial_area: FacialArea {
                topleft_x: 28.43473,
                topleft_y: 35.666706,
                bottomright_x: 97.406876,
                bottomright_y: 134.69205
            }
        },
        landmarks: Landmarks {
            eye_left: Point {
                x: 40.67421,
                y: 78.70622
            },
            eye_right: Point {
                x: 69.57788,
                y: 76.26257
            },
            nose: Point {
                x: 50.147636,
                y: 97.665665
            },
            mouth_right: Point {
                x: 46.10032,
                y: 109.27576
            },
            mouth_left: Point {
                x: 71.982925,
                y: 106.96481
            }
        }
    }
]);
```

Face recognition For face recognition, face-lib utilizes Arcface, a highly accurate algorithm chosen for its exceptional performance in extracting embeddings from detected faces (c.f. Listing 8.3.1). This model was specifically selected and employed throughout this thesis due to its accuracy and robustness in various biometric recognition tasks, as demonstrated in Chapter 3.1.2. To illustrate the straightforward implementation of Arcface within our library, consider the following code example:

```
use face::Pipeline;
use face::img_read;
use face::detection::retinaface::Retinaface;
use face::recognition::arcface::ArcFace;

let mut pipeline = Pipeline::new(
    Box::new(Retinaface::new("./static/detection/retinaface-150x150.tflite", "./static/detection/retinaface_anchors-150x150.json").unwrap()),
    Box::new(ArcFace::new("./static/recognition/arcface.tflite"))
);
let img = img_read!("static/test-images/person.png");
let result = pipeline.calc_embs(&img);
assert_eq!(result.len(), 1);
assert_eq!(result[0], vec![0.2446, 0.135366, 0.6693452, ...]);
```

Visualization Visual feedback is crucial in certain scenarios, particularly during the debugging and development phases of face detection and recognition systems. To facilitate this, face-lib includes visualization capabilities that provide clear, graphical representations of the detection and recognition processes. These visualization tools offer insights into the system performance, allowing developers to:

- Identify and troubleshoot potential issues in face detection
- Verify the accuracy of facial landmark placement
- Understand how different lighting conditions or facial orientations affect the system's performance
- Fine-tune parameters for optimal results

To demonstrate these visualization features, consider the following code example:

```
use face::visualization::Visualization;
use face::detection::fast_detection::Fastdet;

let mut f = Fastdet::new("static/detection/fastdet_320.onnx", 0.7, 320, 240).
    unwrap();
let img = face::img_read!("static/test-images/group.webp");
let res = f.inference(&img).unwrap();
let mut vis = Visualization::new(img);
vis.add_det_results(&res);
```

When executed, this code generates a visual output as illustrated in Figure 8.5.

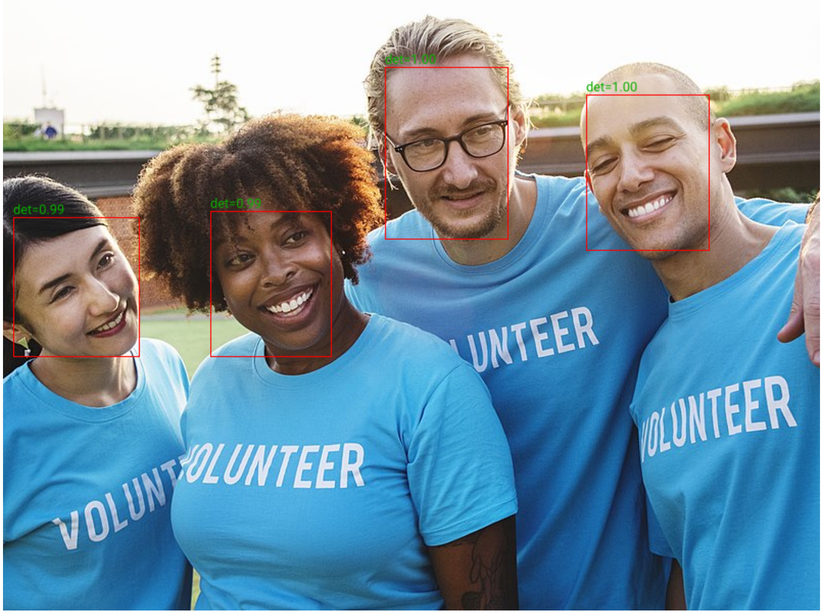


Figure 8.5: Face detection results using face-lib, showing bounding boxes, facial landmarks, and confidence scores for detected faces. Image source: pxphere.com (CC0) https://pxhere.com/en/photo/1438955

8.3.2 sensor-lib

This library is publicly available at https://git.ins.jku.at/proj/digidow/face-lib and in Chapter A.

Source code

This library provides essential functionalities for implementing a Digidow sensor:

- **Registration:** Allows entities to register with the sensor (c.f. Section 8.1.2). Registrations are (optionally) automatically purged after a predefined timeout period to prevent data bloat. Multiple registrations utilizing the same identifier but having different callback URLs are possible. Upon a successful biometric match, all specified callback URLs are simultaneously notified.
- **REST Service:** The library provides a REST interface over a SOCKS proxy to manage registration requests, utilizing the Tor network to ensure secure and anonymous communication. The interface of the sensor library is designed to accept registration requests and includes a heartbeat mechanism to maintain an active connection. The system supports a single route: a POST request to */v1/register*, which requires the inclusion of a verifiable presentation in the request body. Listing 8.3.2 illustrates an example of a verifiable credential included in the VP. This presentation must be signed by a trusted issuing authority to ensure its authenticity and integrity.

```
Object {
    "@context": Array [
        String("https://www.w3.org/2018/credentials/v1"),
        String("https://digidow.eu/v1"),
    ],
    "credentialSubject": Object {
        "datetime": String("2024-08-14T14:32:28UTC"),
        "door-id": String("42"),
        "identity": // embedding; removed for brevity,
        "proposed-verifiers": Array [
            String("verifier.onion"),
        ],
    },
    "expirationDate": String("2024-08-14T14:32:28UTC"),
    "issuanceDate": String("2024-08-14T14:32:28UTC"),
    "issuer": String("Self"),
    "proof": // removed for brevity,
    "type": String("door-opening-data"),
}
```

- **Verifiable Presentations (VPs) [198]:** Utilizes VPs for secure and verifiable exchanges during both registration and sensor push notifications to ensure data integrity and trustworthiness. The sensor has a list of trusted issuing authorities. When the sensor receives a registration, it verifies whether the registration is signed by one of these trusted issuing authorities. This verification is important because it ensures that only credentials from legitimate and recognized sources are accepted, thereby preventing unauthorized or malicious entities from registering. If the registration is not verified as being from a trusted authority, the sensor discards it and informs the PIA accordingly.

- **Sensor push:** Notifies the Personal Identity Agent (PIA) via a registered callback URL when a person is detected by the sensor. We do this in three steps:

  1. The process begins with the creation of a verifiable presentation. This step involves retrieving the verifiable credentials (VCs) specific to the sensor's modality—such as face recognition data, which might include embeddings and related intent information. The code snippet below illustrates the procedure:

```
    // Retrieve verifiable credentials for modality
    let mut vcs = data.into();
    for vc in &mut vcs {
        // Sign them
        vc.sign(&self.bbs_key_pair.secret_key, &self.bbs_key_pair.public_key)?;
    }

    let mut vp = VerifiablePresentation::new(
        vec![
            super::vc::CONTEXT_W3.to_string(),
            super::vc::CONTEXT_DIGIDOW.to_string(),
        ],
        Entity::Single("sensorpush".to_string()),
        vcs,
        Entity::Single(HashMap::new()),
        );
```

  In this step, the modality-specific VCs are first retrieved. These credentials are then individually signed by the sensor using its BBS key pair. The signed VCs ensure that the recipient PIA can transfer the signed credentials to a potential verifier securely, with the guarantee, that it was created by the sensor. Finally, the individually signed VCs are aggregated into a single verifiable presentation (VP).

  2. Once the VP is created, it must be signed to ensure its authenticity and integrity. The VP is signed using the sensor's BBS key pair, as demonstrated in the following code snippet:

```
vp.sign(&self.bbs_key_pair.secret_key, &self.bbs_key_pair.public_key)?;
```

  3. The final step involves sending the signed VP to the PIA via the specified callback URL. This is carried out using an HTTP POST request, which is sent through a Tor socks proxy to ensure privacy and security. The relevant code for this process is as follows:

```
std::thread::spawn(move || {
    debug!("Sending VP to {}: {}", url.onion_address, &json!(vp));

    let proxy = ureq::Proxy::new("socks5://localhost:9050").expect("Valid proxy string");

    let agent = ureq::AgentBuilder::new()
        .timeout(std::time::Duration::from_secs(timeout_in_seconds))
        .proxy(proxy)
        .build();

    match agent
        .post(&format!("{}/v1/sensor-push", url.onion_address))
        .send_json(json!(vp))
    {
        Ok(resp) => {
            info!("PIA received sensor push successfully: {:?}", resp);
        }
        // Error handling removed for brevity
    };
});
```

In this step, a new thread is spawned to handle the transmission of the VP. The VP is serialized into JSON format and sent to the PIA through the specified callback URL using the ureq library. A SOCKS5 proxy is employed to enhance the security of the data transmission. Upon successful delivery, the PIA acknowledges receipt of the VP, as indicated by the log messages.

Digidow utilizes Tor for all communications to ensure network anonymity and enhance security [89]. Tor's anonymizing network helps protect the privacy of both the Personal Identity Agents (PIAs) and the sensors, and furthermore, its onion service makes it easier to reach and communicate securely.

To further bolster the security and trustworthiness of the system, Digidow employs a security architecture based on a hardware root of trust, secure boot mechanisms, and continuous OS integrity verification for the sensor. These security measures ensure that the system operates within a trusted environment before any sensitive data is handled or transmitted.

The hardware root of trust is established through the use of a Trusted Platform Module (TPM), which underpins the secure boot process by verifying each stage of the system's startup. This process ensures that any unauthorized modifications to the system are detected early, preventing compromised systems from functioning. Once the system is operational, continuous integrity checks are performed to ensure that the operating system remains in a known and trusted state, thereby maintaining the integrity of the environment in which the PIAs and sensors operate. These concepts align with broader research on system integrity and attestation in secure environments, as discussed in a previous master thesis on this subject [164].

Furthermore, we implement a timeout mechanism for Tor communications to maintain robust and reliable operations. If a sensor push notification is not completed within the specified timeout, it will be dropped to avoid indefinite delays. Additionally, a heartbeat mechanism is used to keep the connection active and prevent unintended disconnections.

All communications are secured using verifiable presentations (VPs) and verifiable credentials [198]. By signing each VP, we guarantee the authenticity and integrity of the data, ensuring that the recipient can confidently verify that the VP originated from the designated sensor. The sensor stores its cryptographic keys on the device itself, which is protected by full-disk encryption to ensure that the keys remain secure.

The REST service in sensor-lib handles the registration process and accepts and manages registration requests. Registrations are automatically deleted after a predefined period to prevent data bloat and ensure the system remains efficient. This automatic deletion mechanism also enhances security by minimizing the duration for which sensitive biometric data is stored. Additionally, by limiting the retention period, the system makes it more challenging to estimate the number of unique individuals who may have interacted with the sensor, further protecting the privacy of those individuals.

Upon receiving a “sensing received” signal, the library performs a biometric match against all registered identifiers. It then creates a VP for each match and sends the resulting VPs to the specified callback addresses in the registration, ensuring timely and secure delivery of authentication results.

8.3.3 Sensor orchestration

The sensor component integrates the functionalities provided by the face-lib and sensor-lib libraries, orchestrating the overall system operation. There are two main types of sensors in our setup, each tailored to specific deployment scenarios:

1. **Single door sensors:** These sensors are mounted directly above or around individual doors and are responsible for recognizing individuals attempting to gain access. They employ two primary filters to enhance accuracy and security:

   a) **Nose location filter:** This filter ensures that the nose is within a specific image area, allowing the operator to focus the sensor’s detection capability on designated subareas. This is crucial for excluding irrelevant areas such as public walkways and concentrating solely on the entrance area, thereby reducing false triggers and enhancing security.

   b) **Minimum face area filter:** This filter acts as a proxy for intent detection by requiring the person to be within a certain proximity before the system initiates the unlocking process. As the person approaches the door, the size of their face in the image increases. By setting a threshold for the minimum face area, the system can ensure that only individuals close enough to the door will trigger the unlocking mechanism.

2. **Hallway sensors:** These sensors are designed to manage multiple doors within a corridor or hallway. They utilize the 2D coordinates of the doors and individuals’ nose heights to calculate the distance to each door. The system assumes that the closest door within a predefined threshold is the

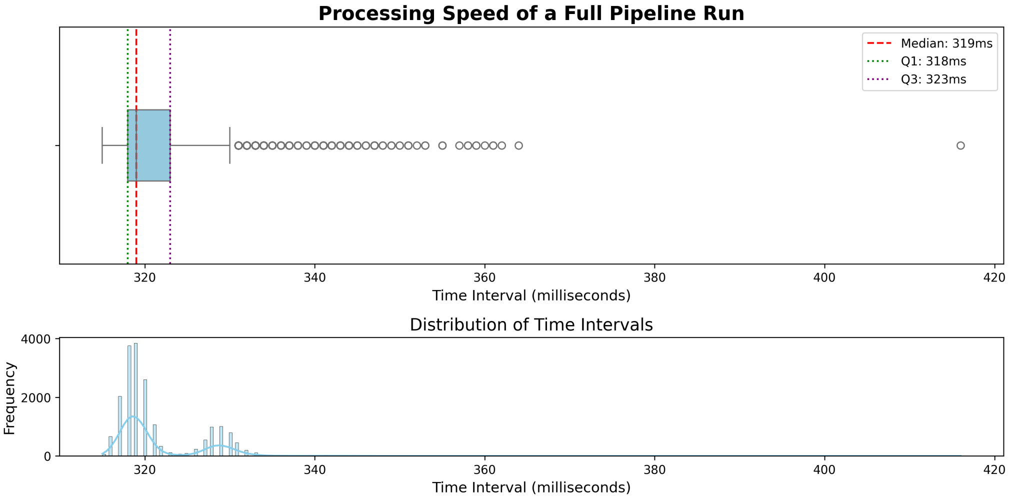


Figure 8.6: Performance analysis of the full face recognition pipeline. (Top) Boxplot showing the distribution of processing times for the complete pipeline, including image acquisition, face detection, feature extraction, and matching against registered embeddings. (Bottom) Probability density function of processing times, highlighting the overall speed distribution of the system.

intended target for access. This setup allows for efficient and accurate detection of individuals' intentions in a more complex environment with multiple access points.

8.4 Results

The living lab prototype successfully demonstrated the practical viability of our decentralized biometric authentication system. Performance testing was conducted on the Jetson Nano hardware platform, processing video streams from 4k and lower-quality cameras across multiple scenarios (c.f. Fig. 8.6).

The results provide evidence for the real-world applicability of the theoretical advancements developed throughout this thesis.

Key findings from our prototype evaluation include:

- **Frame rate:** The system achieved a consistent 3 frames per second (FPS) processing rate across all cameras. This meets the minimum threshold of 3–5 FPS suggested by Stewart et al. [199] for responsive real-time systems.
- **Local processing:** All computations, including face detection, recognition, and authentication logic, were performed entirely on the Jetson Nano without offloading to external GPU servers. This validates the feasibility of edge-based biometric processing, which is crucial for:
  - Energy efficiency: Minimizing power consumption, with the entire pipeline consuming only 10.82 watts, including data transmission and centralized processing.

  - Scalability: Enabling a wide variety of sensors to be deployed without reliance on centralized infrastructure.
  - Privacy enhancement: Minimizing the transmission of sensitive biometric data across networks.

- **Accuracy:** Utilizing the optimized face detection pipeline developed in Chapter 7, the system achieved 98.3 % accuracy on the LFW dataset. This approaches the 99.3 % baseline of unoptimized models, which require over 1.5 minutes to process a single frame. The minimal accuracy trade-off for a significant performance gain (from 90 seconds to 0.33 seconds per frame) demonstrates the effectiveness of our optimization strategies.
- **Network challenges:** The prototype implementation revealed that Tor, while crucial for anonymity, introduced significant latency. Sensor push notifications could take up to 10 seconds to traverse the Tor network, resulting in noticeable delays for users awaiting door access. This highlighted the need to balance security measures with usability considerations in real-world deployments.
- **Connection management:** The prototype exposed the necessity for implementing a heartbeat mechanism to maintain active Tor connections. Without this, connections would prematurely close, disrupting the system's functionality.

These results provide strong empirical evidence for the viability of decentralized, privacy-preserving biometric authentication systems on embedded hardware platforms. They highlight the practical impact of the theoretical improvements developed throughout this thesis, demonstrating their effectiveness in addressing real-world challenges.

The prototype's performance in both the hallway and single-door scenarios further illustrates the system's adaptability to different environmental contexts. This versatility is crucial for widespread adoption across various applications, from secure facility access to border access control.

While these results are highly promising, it is important to acknowledge that further optimization and long-term testing in diverse real-world environments will be necessary to validate the system's robustness and scalability fully. Nonetheless, the current prototype provides a solid foundation for future research and development in decentralized biometric authentication systems.

Chapter 9

Conclusion and outlook

9.1 Conclusion

After the extensive research presented in this thesis, it is time to step back and evaluate how well the work aligns with the broader objectives outlined at the beginning. The overarching aim was to develop a decentralized biometric authentication system that not only enhances privacy but also improves efficiency and scalability. The work presented here has laid down a solid foundation, demonstrating that decentralization is both a viable and necessary evolution for modern, privacy-focused biometric systems.

Central to our research was the focus on developing efficient sensors for biometric authentication. This emphasis stems from the observation, that the sensor represents the first point of interaction and a potential bottleneck. Efficient sensors are important, because in a decentralized architecture, where processing may occur on edge devices rather than centralized servers, resource constraints become a significant consideration. By optimizing sensor efficiency, we can reduce computational demands, lower power consumption, and ultimately make decentralized biometric systems more feasible and widespread.

Our approach to enhancing sensor efficiency was multifaceted, addressing various aspects of the biometric authentication pipeline. We began by examining the fundamental building blocks—the facial features and their representations—and progressively built up to system-level optimizations and a flexible framework for sensor integration.

Our research journey began by challenging conventional wisdom in facial recognition technology. We demonstrated that facial embeddings could be significantly reduced in size while maintaining comparable accuracy levels, a finding that opens new possibilities for biometric systems in resource-constrained environments. Building on this foundation, we proposed embedding fusion techniques that are computationally efficient, even on low-resource devices. As our work progressed, we expanded our focus to system-wide improvements. The development of an efficient face detection pipeline for embedded systems and the introduction of the Domain-Specific Sensor Language (BioDSSL) represent steps towards more adaptable biometric systems. These advancements aim to facilitate easier integration of biometric authentication in various applications, balancing security needs with practical constraints.

The true test of our research came with its real-world implementation. This real-world implementation validated our theoretical constructs and integrated the developed techniques into a functional system. This phase of our research provided practical insights into the challenges of deploying decentralized biometric systems, showcasing the robustness of our approaches and the potential for scalability and privacy enhancements.

Looking ahead, the decentralized framework we have developed offers a potential path for addressing some of the current challenges in biometric authentication. As digital identity and access control continue to evolve, our research provides insights that may be valuable for future developments in the field.

In conclusion, this thesis presents a series of targeted advancements in biometric authentication, with a focus on decentralization and efficiency. By offering solutions to specific challenges and exploring new approaches, we've contributed to the ongoing improvement of biometric systems. As the field continues to evolve, the work presented here serves as a stepping stone towards more secure, efficient, and privacy-aware biometric authentication methods.

These contributions collectively advance the field of biometric authentication, particularly in the context of decentralized and privacy-preserving systems.

9.2 Future work

As with any journey, the path laid out in this thesis opens up new directions for exploration, offering exciting prospects for future research and development.

One immediate direction involves the completion and publication of a detailed survey, grounded in the foundational work laid out in Section 2.1. This survey aims to consolidate the insights gained, offering a comprehensive overview that can serve as a reference point for future research.

Another promising avenue lies in enhancing biometric recognition systems by refining the training of models, specifically through the implementation of a more constrained output or embedding layer. Such an approach is expected to streamline the inference process, reducing the dependency on extensive post-processing, which in turn could lead to improvements in both the accuracy and speed of these systems.

Additionally, the innovative method proposed in this thesis for aggregating embeddings from different neural networks is ripe for further investigation. Future research could explore the application of similar aggregation techniques across various biometric modalities or within multi-modal systems. This holds the potential to create more robust and versatile recognition systems, capable of delivering higher performance in diverse real-world scenarios.

Bibliography

[1] 2016. *Procedia Computer Science*, 85, (January 2016), 109–116. ISSN: 1877-0509. DOI: 10.1016/j.procs.2016.05.187.

[2] Goutham Reddy Alavalapati et al. 2016. Biometric authentication using near infrared hand vein pattern with adaptive threshold technique. In *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*. IEEE, pp. 229–234.

[3] Ghazel Albakri and Sharifa Alghowinem. 2019. The effectiveness of depth data in liveness face authentication using 3D sensor cameras. *Sensors*, 19, 8, 1928.

[4] A Tahseen Ali, Hasanen S Abdullah, and Mohammad N Fadhil. 2021. Voice recognition system using machine learning techniques. *Materials Today: Proceedings*, 1–7.

[5] Kamran Ali, Alex X Liu, Wei Wang, and Muhammad Shahzad. 2015. Keystroke recognition using wifi signals. In *Proceedings of the 21st annual international conference on mobile computing and networking*, pp. 90–102.

[6] Aishat Aloba, Sarah Morrison-Smith, Aaliyah Richlen, Kimberly Suarez, Yu-Peng Chen, Jaime Ruiz, and Lisa Anthony. 2023. Multi-modal User Authentication in Smart Environments: Survey of User Attitudes. *arXiv preprint arXiv:2305.03699*.

[7] J Andrews, A Vakil, and J Li. 2020. Biometric authentication and stationary detection of human subjects by deep learning of passive infrared (PIR) sensor data. In *2020 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*. IEEE, pp. 1–6.

[8] K Annapurani, C Malathy, Hardeep Singh, and Dhiraj J Rathod. 2016. Face Authentication System of Thermal Image with Gabor Filter. *Indian Journal of Science and Technology*.

[9] Patricia Arias-Cabarcos, Christian Krupitzer, and Christian Becker. 2019. A survey on adaptive authentication. *ACM Computing Surveys (CSUR)*, 52, 4, 1–30.

[10] Sercan Ö Arık, Mike Chrzanowski, Adam Coates, Gregory Diamos, Andrew Gibiansky, Yongguo Kang, Xian Li, John Miller, Andrew Ng, Jonathan Raiman, et al. 2017. Deep voice: Real-time neural text-to-speech. In *International conference on machine learning*. PMLR, pp. 195–204.

[11] Muhammad Arsalan, Hyung Gil Hong, Rizwan Ali Naqvi, Min Beom Lee, Min Cheol Kim, Dong Seop Kim, Chan Sik Kim, and Kang Ryoung Park. 2017. Deep learning-based iris segmentation for iris recognition in visible light environment. *Symmetry*, 9, 11, 263.

[12] S Athira and OV Ramana Murthy. 2018. Face authentication using thermal imaging. In *Computational Vision and Bio Inspired Computing*. Springer, pp. 1006–1014.

[13] Tarryn Balsdon, Stephanie Summersby, Richard I Kemp, and David White. 2018. Improving face identification with specialist teams. *Cognitive Research: Principles and Implications*, 3, 1, 1–13.

[14] Ankan Bansal, Anirudh Nanduri, Carlos D Castillo, Rajeev Ranjan, and Rama Chellappa. 2017. Umdfaces: An annotated face dataset for training deep networks. In *IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, pp. 464–473. DOI: 10.1109/BTAS.2017.8272731.

[15] F Battaglia, Giancarlo Iannizzotto, and L Lo Bello. 2014. A biometric authentication system based on face recognition and rfid tags. *Mondo Digitale*, 13, 49, 340–346.

[16] John Berry and David A Stoney. 2001. The history and development of fingerprinting. *Advances in fingerprint Technology*, 2, 13–52.

[17] Samarth Bharadwaj, Mayank Vatsa, and Richa Singh. 2014. Biometric quality: a review of fingerprint, iris, and face. *EURASIP journal on Image and Video Processing*, 2014, 1, 1–28.

[18] Ramon Blanco-Gonzalo, Oscar Miguel-Hurtado, Chiara Lunerti, Richard M Guest, Barbara Corsetti, Elakkiya Ellavarason, and Raul Sanchez-Reillo. 2019. Biometric systems interaction assessment: the state of the art. *IEEE Transactions on Human-Machine Systems*, 49, 5, 397–410.

[19] Jorge Blasco, Thomas M Chen, Juan Tapiador, and Pedro Peris-Lopez. 2016. A survey of wearable biometric recognition systems. *ACM Computing Surveys (CSUR)*, 49, 3, 1–35.

[20] Ghoroub Talal Bostaji and Emad Sami Jaha. 2023. Fine-Grained Soft Ear Biometrics for Augmenting Human Recognition. *Computer Systems Science & Engineering*, 47, 2.

[21] Jason Boyd, Muhammad Fahim, and Oluwafemi Olukoya. 2023. Voice spoofing detection for multiclass attack classification using deep learning. *Machine Learning with Applications*, 14, 100503.

[22] Hervé Bredin and Antoine Laurent. 2021. End-to-end speaker segmentation for overlap-aware resegmentation. *arXiv preprint arXiv:2104.04045*.

[23] Sandrine Brognaux and Thomas Drugman. 2015. HMM-based speech segmentation: Improvements of fully automatic approaches. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24, 1, 5–15.

[24] Ulrich Burgbacher, Manuel Prätorius, and Klaus Hinrichs. 2014. A behavioral biometric challenge and response approach to user authentication on smartphones. In *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, pp. 3328–3335.

[25] Minjie Cai, Feng Lu, and Yoichi Sato. 2020. Generalizing hand segmentation in egocentric videos with uncertainty-guided model adaptation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pp. 14392–14401.

[26] Kai Cao and Anil K Jain. 2015. Latent orientation field estimation via convolutional neural network. In *2015 International Conference on Biometrics (ICB)*. IEEE, pp. 349–356.

[27] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. 2018. VGGFace2: A Dataset for Recognising Faces across Pose and Age. In *13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*. IEEE, pp. 67–74. DOI: 10.1109/FG.2018.00020.

[28] V Vanitha Carmel and D Akila. 2020. A survey on biometric authentication systems in cloud to combat identity theft. *Journal of Critical Reviews*, 7, 03, 540–547.

[29] Veena Chandran and Philomina Simon. 2019. Review on dental image based biometric system. In *Proceedings of the Third International Conference on Advanced Informatics for Computing Research*, pp. 1–6.

[30] Parag Chatterjee and Asoke Nath. 2015. Biometric authentication for UID-based smart and ubiquitous services in India. In *2015 Fifth International Conference on Communication Systems and Network Technologies*. IEEE, pp. 662–667.

[31] Yi Chen, Sarat C Dass, and Anil K Jain. 2005. Fingerprint quality indices for predicting authentication performance. In *International conference on audio-and video-based biometric person authentication*. Springer, pp. 160–170.

[32] Megha Chhabra, Kiran Kumar Ravulakollu, Manoj Kumar, Abhay Sharma, and Anand Nayyar. 2023. Improving automated latent fingerprint detection and segmentation using deep convolutional neural network. *Neural Computing and Applications*, 35, 9, 6471–6497.

[33] Hyunsoek Choi, Hyeyoung Park, et al. 2015. A multimodal user authentication system using faces and gestures. *BioMed research international*, 2015.

[34] Swati K. Choudhary and Ameya K. Naik. 2019. Multimodal Biometric Authentication with Secured Templates — A Review. In *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*. (April 2019), pp. 1062–1069. DOI: 10.1109/ICOEI.2019.8862563.

[35] Aruni Roy Chowdhury, Tsung-Yu Lin, Subhransu Maji, and Erik Learned-Miller. 2016. One-to-many face recognition with bilinear cnns. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, pp. 1–9.

[36] Simon A Cole et al. 2009. *Suspect identities: A history of fingerprinting and criminal identification*. Harvard University Press.

[37] Vincenzo Conti, C Militello, and S Vitabile. 2017. Biometric authentication overview: a fingerprint recognition sensor description. *Int J Biosen Bioelectron*, 2, 1, 26–31.

[38] Rodrigo Colnago Contreras, Monique Simplicio Viana, and Rodrigo Capobianco Guido. 2023. An Experimental Analysis on Mapping Strategies for Cepstral Coefficients Multi-projection in Voice Spoofing Detection Problem. In *International Conference on Artificial Intelligence and Soft Computing*. Springer, pp. 291–306.

[39] D. Crocker and P. Overell. 2008. Augmented BNF for Syntax Specifications: ABNF. RFC 5234. Internet Engineering Task Force, (January 2008). https://www.rfc-editor.org/rfc/rfc5234.txt.

[40] Adam Czajka. 2015. Pupil dynamics for iris liveness detection. *IEEE Transactions on Information Forensics and Security*, 10, 4, 726–735.

[41] Adam Czajka and Pawel Bulwan. 2013. Biometric verification based on hand thermal images. In *2013 International Conference on Biometrics (ICB)*. IEEE, pp. 1–6.

[42] Naser Damer, Jonas Henry Grebe, Cong Chen, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. 2020. The effect of wearing a mask on face recognition performance: an exploratory study. In *2020 International Conference of the Biometrics Special Interest Group (BIOSIG)*. IEEE, pp. 1–6.

[43] Antitza Dantcheva, Petros Elia, and Arun Ross. 2015. What else does your biometric data reveal? A survey on soft biometrics. *IEEE Transactions on Information Forensics and Security*, 11, 3, 441–467.

[44] Shaveta Dargan and Munish Kumar. 2020. A comprehensive survey on the biometric recognition systems based on physiological and behavioral modalities. *Expert Systems with Applications*, 143, 113114.

[45] Shaveta Dargan and Munish Kumar. 2020. A comprehensive survey on the biometric recognition systems based on physiological and behavioral modalities. *Expert Systems with Applications*, 143, (April 2020), 113114. ISSN: 0957-4174. DOI: 10.1016/j.eswa.2019.113114.

[46] Ashok Kumar Das, Sherali Zeadally, and Mohammad Wazid. 2017. Lightweight authentication protocols for wearable devices. *Computers & Electrical Engineering*, 63, 196–208.

[47] Priyanka Das, Joseph McFiratht, Zhaoyuan Fang, Aidan Boyd, Ganghee Jang, Amir Mohammadi, Sandip Purnapatra, David Yambay, Sébastien Marcel, Mateusz Trokielewicz, et al. 2020. Iris liveness detection competition (livdet-iris)-the 2020 edition. In *2020 IEEE international joint conference on biometrics (IJCB)*. IEEE, pp. 1–9.

[48] John Daugman. 2009. How iris recognition works. In *The essential guide to image processing*. Elsevier, pp. 715–739.

[49] Hugh Davson. 1990. *Physiology of the Eye*. Bloomsbury Publishing.

[50] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. 2020. Retinaface: Single-shot Multi-Level Face Localisation in the Wild. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5202–5211. DOI: 10.1109/CVPR42600.2020.00525.

[51] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. 2019. ArcFace: Additive Angular Margin Loss for Deep Face Recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4685–4694. DOI: 10.1109/CVPR.2019.00482.

[52] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. 2019. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4690–4699.

[53] Jiankang Deng, Jia Guo, Yuxiang Zhou, Jinke Yu, Irene Kotsia, and Stefanos Zafeiriou. 2019. Retinaface: Single-stage dense face localisation in the wild. *arXiv preprint arXiv:1905.00641*.

[54] Department of Defense. 2000. Human engineering design data digest. Accessed April 3, 2023. https://apps.dtic.mil/sti/pdfs/ADA467401.pdf. (2000).

[55] Arindam Dutta, Rohit Lal, Dripta S Raychaudhuri, Calvin-Khang Ta, and Amit K Roy-Chowdhury. 2024. POISE: Pose Guided Human Silhouette Extraction under Occlusions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 6153–6163.

[56] Simon Eberz, Giulio Lovisotto, Andrea Patane, Marta Kwiatkowska, Vincent Lenders, and Ivan Martinovic. 2018. When your fitness tracker betrays you: Quantifying the predictability of biometric features across contexts. In *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, pp. 889–905.

[57] Faouzia Ennaama, Khalid Benhida, Ahmed Boulahoual, Ahmed Bentajer, Hedabou Mustapha, and Said Elfezazi. 2019. Comparative and analysis study of biometric systems. *Journal of Theoretical and Applied Information Technology*, 97, 12.

[58] europa.eu. 2023. Entry/Exit System (EES). Accessed April 3, 2023. https://home-affairs.ec.europa.eu/policies/schengen-borders-and-visa/smart-borders/entry-exit-system_en. (2023).

[59] Jude Ezeobiejesi and Bir Bhanu. 2017. Latent fingerprint image segmentation using deep neural network. *Deep Learning for Biometrics*, 83–107.

[60] Zhenyu Fang, Jinchang Ren, Stephen Marshall, Huimin Zhao, Zheng Wang, Kaizhu Huang, and Bing Xiao. 2020. Triple loss for hard face detection. *Neurocomputing*, 398, 20–30.

[61] Sachin Sudhakar Farfade, Mohammad J Saberian, and Li-Jia Li. 2015. Multi-view face detection using deep convolutional neural networks. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, pp. 643–650. DOI: 10.48550/arXiv.1502.02766.

[62] Yuantao Feng, Shiqi Yu, Hanyang Peng, Yan-Ran Li, and Jianguo Zhang. 2022. Detect Faces Efficiently: A Survey and Evaluations. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 4, 1, 1–18. DOI: 10.1109/TBIOM.2021.3120412.

[63] Julian Fierrez-Aguilar, Javier Ortega-Garcia, Joaquin Gonzalez-Rodriguez, and Josef Bigun. 2005. Discriminative multimodal biometric authentication based on quality measures. *Pattern recognition*, 38, 5, 777–779.

[64] Rainhard Dieter Findling and Rene Mayrhofer. 2013. Towards pan shot face unlock: Using biometric face information from different perspectives to unlock mobile devices. *International Journal of Pervasive Computing and Communications*.

[65] Andrian Firmansyah, Tien Fabrianti Kusumasari, and Ekky Novriza Alam. 2023. Comparison of face recognition accuracy of ArcFace, FaceNet and FaceNet512 models on deepface framework. In *2023 International conference on computer science, information technology and engineering (ICCoSITE)*. IEEE, pp. 535–539.

[66] Simon Fong, Yan Zhuang, and Iztok Fister. 2013. A biometric authentication model using hand gesture images. *Biomedical engineering online*, 12, 1, 1–18.

[67] Shilpa Garg, Sumit Mittal, Pardeep Kumar, and Vijay Anant Athavale. 2020. DeBNet: multilayer deep network for liveness detection in face recognition system. In *2020 7th International Conference on Signal Processing and Integrated Networks (SPIN)*. IEEE, pp. 1136–1141.

[68] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. 2021. Datasheets for datasets. *Communications of the ACM*, 64, 12, 86–92.

[69] Emad Sami Jaha Ghoroub Talal Bostaji. 2023. Fine-Grained Soft Ear Biometrics for Augmenting Human Recognition. *Computer Systems Science and Engineering*, 47, 2, 1571–1591. DOI: 10.32604/csse.2023.039701. http://www.techscience.com/csse/v47n2/53647.

[70] A Jay Goldstein, Leon D Harmon, and Ann B Lesk. 1971. Identification of human faces. *Proceedings of the IEEE*, 59, 5, 748–760.

[71] Sixue Gong, Yichu Shi, Nathan D Kalka, and Anil K Jain. 2019. Video face recognition: Component-wise feature aggregation network (c-fan). In *2019 International Conference on Biometrics (ICB)*. IEEE, pp. 1–8.

[72] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. 2016. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*. Springer, pp. 87–102.

[73] Mohamed Hammad, Yashu Liu, and Kuanquan Wang. 2018. Multimodal biometric authentication systems using convolution neural network based on different level fusion of ECG and fingerprint. *IEEE Access*, 7, 26527–26542.

[74] Md Khaled Hasan, Md Shamim Ahsan, SH Shah Newaz, and Gyu Myoung Lee. 2021. Human face detection techniques: A comprehensive review and future research directions. *Electronics*, 10, 19, 2354.

[75] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[76] Brook Heisler. 2014. criterion.rs: Statistics-driven benchmarking library for Rust. https://github.com/bheisler/criterion.rs. (2014).

[77] S Hemalatha. 2020. A systematic review on Fingerprint based Biometric Authentication System. In *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*. IEEE, pp. 1–4.

[78] **Hofer, Philipp**. 2021. Analysis of state-of-the-art off-the-shelve face recognition pipelines. Technical report. Johannes Kepler University Linz, Institute of Networks and Security, Christian Doppler Laboratory for Private Digital Authentication in the Physical World, (March 2021). https://www.digidow.eu/publications/2021-hofer-tr-analysisfacerecognitionpipelines/Hofer_2021_AnalysisFaceRecognitionPipelines.pdf.

[79] **Hofer, Philipp**. 2023. Dezentrale Gesichtserkennung. *OCG Journal*, 48, 1, (April 2023), 14–15. https://www.ocg.at/sites/ocg.at/files/medien/pdfs/OJ2023-01.pdf.

[80] **Hofer, Philipp**. 2022. Die Bedeutung verschiedener Gesichtsteile für Gesichtserkennung und dessen Zusammenführung. In *IKT-Sicherheitskonferenz 2022*. Vienna, Austria, (September 2022). https://www.digidow.eu/publications/2022-hofer-iktsicherheitskonferenz/Hofer_2022_IKTSicherheitskonferenz2022_Poster.pdf.

[81] **Hofer, Philipp**. 2021. Face recognition: Increase accuracy by filtering images with heuristics. Technical report. Johannes Kepler University Linz, Institute of Networks and Security, Christian Doppler Laboratory for Private Digital Authentication in the Physical World, (July 2021). https://www.digidow.eu/publications/2021-hofer-tr-increasefacerecognitionaccuracy/Hofer_2021_IncreaseFaceRecognitionAccuracy.pdf.

[82] **Hofer, Philipp**, Michael Roland, René Mayrhofer, and Philipp Schwarz. 2023. Optimizing Distributed Face Recognition Systems through Efficient Aggregation of Facial Embeddings. *Advances in Artificial Intelligence and Machine Learning*, 3, 1, (February 2023), 693–711. DOI: 10.54364/AAIML.2023.1146.

[83] **Hofer, Philipp**, Michael Roland, Philipp Schwarz, and René Mayrhofer. 2023. Efficient Aggregation of Face Embeddings for Decentralized Face Recognition Deployments. In *Proceedings of the 9th International Conference on Information Systems Security and Privacy (ICISSP 2023)*. SciTePress, Lisbon, Portugal, (February 2023), pp. 279–286. DOI: 10.5220/0011599300003405.

[84] **Hofer, Philipp**, Michael Roland, Philipp Schwarz, and René Mayrhofer. 2022. Efficient aggregation of face embeddings for decentralized face recognition deployments (extended version). (December 2022). https://arxiv.org/abs/2212.10108.

[85] **Hofer, Philipp**, Michael Roland, Philipp Schwarz, Martin Schwaighofer, and René Mayrhofer. 2021. Importance of different facial parts for face detection networks. In *2021 9th IEEE International Workshop on Biometrics and Forensics (IWBF)*. IEEE, Rome, Italy, (May 2021), pp. 1–6. DOI: 10.1109/IWBF50991.2021.9465087.

[86] **Hofer, Philipp**, Philipp Schwarz, Michael Roland, and René Mayrhofer. 2024. BioDSSL: A Domain Specific Sensor Language for global, distributed, biometric identification systems. In *12th IEEE International Conference on Intelligent Systems (IEEE IS 2024)*. IEEE, Golden Sands, Bulgaria, (August 2024).

[87] **Hofer, Philipp**, Philipp Schwarz, Michael Roland, and René Mayrhofer. 2023. Face to Face with Efficiency: Real-Time Face Recognition Pipelines on Embedded Devices. In *21st International Conference on Advances in Mobile Computing & Multimedia Intelligence (MoMM 2023)*. ACM, Bali, Indonesia, (December 2023).

[88] **Hofer, Philipp**, Philipp Schwarz, Michael Roland, and René Mayrhofer. 2024. Shrinking embeddings, not accuracy: Performance-Preserving Reduction of Facial Embeddings for Complex Face Verification Computations. In *14th International Conference on Pattern Recognition Systems (ICPRS 2024)*. IEEE, London, UK, (July 2024).

[89] Tobias Höller. 2022. *A Privacy Preserving Networking Approach for Distributed Digital Identity Systems*. PhD thesis. Johannes Kepler University Linz, Institute of Networks and Security, Linz, Austria, (October 2022), 153 pages.

[90] Seng Chun Hoo and Haidi Ibrahim. 2019. Biometric-based attendance tracking system for education sectors: A literature survey on hardware requirements. *Journal of Sensors*, 2019.

[91] [n. d.] https://github.com/AIZOOTech/FaceMaskDetection.

[92] https://spec.torproject.org/tor-spec/preliminaries.html?highlight=msg-len%20preliminaries#msg-len. 2024. Tor specifications: Message lengths. Accessed May 5, 2024. (2024).

[93] Yang Hu, Konstantinos Sirlantzis, and Gareth Howells. 2016. Iris liveness detection using regional features. *Pattern Recognition Letters*, 82, 242–250.

[94] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. 2007. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. Technical report 07-49. University of Massachusetts, Amherst, (October 2007).

[95] Alberto Ibarrondo, Hervé Chabanne, and Melek Önen. 2022. Funshade: Functional secret sharing for two-party secure thresholded distance evaluation. *Cryptology ePrint Archive*.

[96] Aderonke Justina Ikuomola. 2015. Fingerprint-based authentication system for time and attendance management. *British Journal of Mathematics & Computer Science*, 5, 6, 735.

[97] D Jagadiswary and D Saraswady. 2016. Biometric authentication using fused multimodal biometric. *Procedia Computer Science*, 85, 109–116.

[98] Anil K Jain, Sarat C Dass, and Karthik Nandakumar. 2004. Can soft biometric traits assist user recognition? In *Biometric technology for human identification*. Volume 5404. Spie, pp. 561–572.

[99] Anil K Jain, Sarat C Dass, and Karthik Nandakumar. 2004. Soft biometric traits for personal recognition systems. In *International conference on biometric authentication*. Springer, pp. 731–738.

[100] Zhe Jin, Meng-Hui Lim, Andrew Beng Jin Teoh, Bok-Min Goi, and Yong Haur Tay. 2016. Generating Fixed-Length Representation From Minutiae Using Kernel Methods for Fingerprint Authentication. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46, 10, 1415–1428. DOI: 10.1109/TSMC.2015.2499725.

[101] Hanyung Jung, Soobin Sim, and Hyunkoo Lee. 2023. Biometric authentication security enhancement under quantum dot light-emitting diode display via fingerprint imaging and temperature sensing. *Scientific Reports*, 13, 1, 794.

[102] Tarun K Kanakam, Ajith Jubilson, Brahmini Emani, Marthala Anuhya, Sneha Sighakolli, Vandana Chintala, Kishan Vanamala, Deepak Kadiri, Kushal Nayineni, and Paneerselvam Dhanavanthini. 2023. A concise survey on biometric recognition methods. *International Journal of Computing and Digital Systems*, 14, 1, 1–1.

[103] Byeongkeun Kang, Kar-Han Tan, Nan Jiang, Hung-Shuo Tai, Daniel Tretter, and Truong Nguyen. 2017. Hand segmentation for hand-object interaction from depth map. In *2017 IEEE global conference on signal and information processing (GlobalSIP)*. IEEE, pp. 259–263.

[104] Nima Karimian, Paul A Wortman, and Fatemeh Tehranipoor. 2016. Evolving authentication design considerations for the internet of biometric things (IoBT). In *Proceedings of the eleventh IEEE/ACM/IFIP international conference on hardware/software codesign and system synthesis*, pp. 1–10.

[105] M Killioğlu, M Taşkiran, and N Kahraman. 2017. Anti-spoofing in face recognition with liveness detection using pupil tracking. In *2017 IEEE 15th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*. IEEE, pp. 000087–000092.

[106] Minchul Kim, Anil K Jain, and Xiaoming Liu. 2022. Adaface: Quality adaptive margin for face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18750–18759.

[107] Davis E King. 2009. Dlib-ml: A machine learning toolkit. *The Journal of Machine Learning Research*, 10, 1755–1758.

[108] Medikonda Asha Kiran, Padmatti Yogeshwari, Kosuru Viswa Bhavani, and Thudumu Ramya. 2018. Biometric authentication: a holistic review. In *2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), 2018 2nd International Conference on*. IEEE, pp. 428–433.

[109] Sheenam Kochar, Er Amarinder Kaur, and Banur SVIET. [n. d.] THE SURVEY OF MULTI-MODEL BIOMETRIC AUTHENTICATION SYSTEM DESIGN BASED ON HAAR WAVELETS AND SUPPORT VECTOR MACHINE.

[110] Ashu Kumar, Amandeep Kaur, and Munish Kumar. 2019. Face detection techniques: a review. *Artificial Intelligence Review*, 52, 927–948.

[111] Ashu Kumar, Munish Kumar, and Amandeep Kaur. 2021. Face detection in still images under occlusion and non-uniform illumination. *Multimedia Tools and Applications*, 80, 14565–14590. DOI: 10.1007/s11042-020-10457-9.

[112] Kunal Kumar and Mohammed Farik. 2016. A review of multimodal biometric authentication systems. *Int. J. Sci. Technol. Res*, 5, 12, 5–9.

[113] Ruggero Donida Labati, Angelo Genovese, Enrique Muñoz, Vincenzo Piuri, Fabio Scotti, and Gianluca Sforza. 2016. Biometric recognition in automated border control: a survey. *ACM Computing Surveys (CSUR)*, 49, 2, 1–39.

[114] Oleksandr Lavrynenko, Alla Pinchuk, Hanna Martyniuk, Andrii Fesenko, Stanislav Yarotsky, and Marek Aleksander. 2023. Remote Voice User Verification System for Access to IoT Services Based on 5G Technologies. In *2023 IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*. Volume 1. IEEE, pp. 1042–1048.

[115] Zhihang Li, Xu Tang, Junyu Han, Jingtuo Liu, and Ran He. 2019. PyramidBox++: High Performance Detector for Finding Tiny Face. (2019). DOI: 10.1904/arXiv.1904.00386. arXiv: 1904.00386 [cs.CV].

[116] Sheng Lian, Zhiming Luo, Zhun Zhong, Xiang Lin, Songzhi Su, and Shaozi Li. 2018. Attention guided U-Net for accurate iris segmentation. *Journal of Visual Communication and Image Representation*, 56, 296–304.

[117] Tailin Liang, John Glossner, Lei Wang, Shaobo Shi, and Xiaotong Zhang. 2021. Pruning and quantization for deep neural network acceleration: A survey. *Neurocomputing*, 461, 370–403.

[118] Chi-Wei Lien and Sudip Vhaduri. 2023. Challenges and Opportunities of Biometric User Authentication in the Age of IoT: A Survey. *ACM Computing Surveys*.

[119] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. 2017. Feature Pyramid Networks for Object Detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 936–944. DOI: 10.1109/CVPR.2017.106.

[120] Linzaer. 2019. 1MB lightweight face detection model. https://github.com/Linzaer/Ultra-Light-Fast-Generic-Face-Detector-1MB. (2019).

[121] Chuncheng Liu. 2019. Multiple social credit systems in China. *Economic Sociology: The European Electronic Newsletter*, 21, 1, 22–32.

[122] Tiantian Liu, Feng Lin, Chao Wang, Chenhan Xu, Xiaoyu Zhang, Zhengxiong Li, Wenyao Xu, Ming-Chun Huang, and Kui Ren. 2023. WavoID: Robust and Secure Multi-modal User Identification via mmWave-voice Mechanism. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pp. 1–15.

[123] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. 2017. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 212–220.

[124] Zhaoxiang Liu, Huan Hu, Jinqiang Bai, Shaohua Li, and Shiguo Lian. 2019. Feature aggregation network for video face recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*.

[125] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3730–3738.

[126] Eduardo Garea Llano, Mireya Saraí García Vázquez, Juan M. Colores Vargas, Luis M. Zamudio Fuentes, and Alejandro A. Ramírez Acosta. 2018. Optimized robust multi-sensor scheme for simultaneous video and image iris recognition. *Pattern Recognition Letters*, 101, (January 2018), 44–51. ISSN: 0167-8655. DOI: 10.1016/j.patrec.2017.11.012.

[127] Giulio Lovisotto, Henry Turner, Simon Eberz, and Ivan Martinovic. 2020. Seeing red: PPG biometrics using smartphone cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 818–819.

[128] Mark Maguire. 2009. The birth of biometric security. *Anthropology today*, 25, 2, 9–14.

[129] Manmeet Mahinderjit Singh, Ke Wan Ching, and Asrulnizam Abd Manaf. 2020. A novel out-of-band biometrics authentication scheme for wearable devices. *International Journal of Computers and Applications*, 42, 6, 589–601.

[130] Giosuè Cataldo Marinò, Alessandro Petrini, Dario Malchiodi, and Marco Frasca. 2021. Compact representations of convolutional neural networks via weight pruning and quantization. *arXiv preprint arXiv:2108.12704*.

[131] Magdin Martin, Koprda Štefan, and Ferenczy L'ubor. 2018. Biometrics authentication of fingerprint with using fingerprint reader and microcontroller Arduino. *Telkomnika (Telecommunication Computing Electronics and Control)*, 16, 2, 755–765.

[132] René Mayrhofer, Michael Roland, and Tobias Höller. 2020. Poster: Towards an Architecture for Private Digital Authentication in the Physical World. In *Network and Distributed System Security Symposium (NDSS Symposium 2020), Posters*. San Diego, CA, USA, (February 2020).

[133] René Mayrhofer, Michael Roland, and Tobias Höller. 2020. Poster: Towards an Architecture for Private Digital Authentication in the Physical World. In *Network and Distributed System Security Symposium (NDSS Symposium 2020), Posters*. San Diego, CA, USA, (February 2020).

[134] Lukas Mecke, Ken Pfeuffer, Sarah Prange, and Florian Alt. 2018. Open sesame! user perception of physical, biometric, and behavioural authentication concepts to open doors. In *Proceedings of the 17th international conference on mobile and ubiquitous multimedia*, pp. 153–159.

[135] Joanna Phillips Melancon and Vassilis Dalakas. 2018. Consumer social voice in the age of social media: Segmentation profiles and relationship marketing strategies. *Business Horizons*, 61, 1, 157–167.

[136] Qiang Meng, Shichao Zhao, Zhida Huang, and Feng Zhou. 2021. Magface: A universal representation for face recognition and quality assessment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14225–14234.

[137] Zuheng Ming, Muriel Visani, Muhammad Muzzamil Luqman, and Jean-Christophe Burie. 2020. A survey on anti-spoofing methods for facial recognition with rgb cameras of generic consumer devices. *Journal of imaging*, 6, 12, 139.

[138] Ryogo Miyazaki, Kazuya Sasaki, Norimichi Tsumura, and Keita Hirai. 2022. Hand authentication from RGB-D video based on deep neural network. *Electronic Imaging*, 34, 1–5.

[139] Daniel Morgan and William Krouse. 2005. Biometric identifiers and border security: 9/11 Commission recommendations and related issues. In Congressional Information Service, Library of Congress.

[140] mos.ru. 2023. The Face Pay system for fare payment was launched at all metro stations. Accessed February 2, 2023. https://www.mos.ru/news/item/97579073/. (2023).

[141] Valerio Mura, Giulia Orrù, Roberto Casula, Alessandra Sibiriu, Giulia Loi, Pierluigi Tuveri, Luca Ghiani, and Gian Luca Marcialis. 2018. LivDet 2017 fingerprint liveness detection competition 2017. In *2018 international conference on biometrics (ICB)*. IEEE, pp. 297–302.

[142] Kamal Nasrollahi and Thomas B Moeslund. 2008. Face quality assessment system in video sequences. In *Biometrics and Identity Management: First European Workshop, BIOID 2008, Roskilde, Denmark, May 7-9, 2008. Revised Selected Papers 1*. Springer, pp. 10–18.

[143] Ryota Natsume, Shunsuke Saito, Zeng Huang, Weikai Chen, Chongyang Ma, Hao Li, and Shigeo Morishima. 2019. Siclope: Silhouette-based clothed people. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4480–4490.

[144] Natalia Neverova, Christian Wolf, Graham W Taylor, and Florian Nebout. 2015. Hand segmentation with structured convolutional learning. In *Computer Vision–ACCV 2014: 12th Asian Conference on Computer Vision, Singapore, Singapore, November 1-5, 2014, Revised Selected Papers, Part III 12*. Springer, pp. 687–702.

[145] Mei L Ngan, Patrick J Grother, and Kayee K Hanaoka. 2020. Ongoing face recognition vendor test (frvt) part 6b: Face recognition accuracy with face masks using post-covid-19 algorithms.

[146] Koichiro Niinuma, Unsang Park, and Anil K Jain. 2010. Soft biometric traits for continuous user authentication. *IEEE Transactions on information forensics and security*, 5, 4, 771–780.

[147] Mark S Nixon, Paulo L Correia, Kamal Nasrollahi, Thomas B Moeslund, Abdenour Hadid, and Massimo Tistarelli. 2015. On soft biometrics. *Pattern Recognition Letters*, 68, 218–230.

[148] Rodrigo Frassetto Nogueira, Roberto de Alencar Lotufo, and Rubens Campos Machado. 2016. Fingerprint liveness detection using convolutional neural networks. *IEEE transactions on information forensics and security*, 11, 6, 1206–1213.

[149] Henry Friday Nweke, Ying Wah Teh, Ghulam Mujtaba, Uzoma Rita Alo, and Mohammed Ali Al-garadi. 2019. Multi-sensor fusion based on multiple classifier systems for human activity identification. *Human-centric Computing and Information Sciences*, 9, 1, 1–44.

[150] Uzoma I Oduah, Ifeanyichukwu F Kevin, Daniel O Oluwole, and Josephat U Izunobi. 2021. Towards a high-precision contactless fingerprint scanner for biometric authentication. *Array*, 11, 100083.

[151] Obi Ogbanufe and Dan J Kim. 2018. Comparing fingerprint-based biometrics authentication versus traditional authentication methods for e-payment. *Decision Support Systems*, 106, 1–14.

[152] Kennedy Okokpujie, Etinosa Noma-Osaghae, Olatunji Okesola, Osemwegie Omoruyi, Chinonso Okereke, Samuel John, and Imhade P Okokpujie. 2019. Fingerprint biometric authentication based point of sale terminal. In *Information Science and Applications 2018: ICISA 2018*. Springer, pp. 229–237.

[153] Muhtahir O Oloyede and Gerhard P Hancke. 2016. Unimodal and multimodal biometric sensing systems: a review. *IEEE access*, 4, 7532–7555.

[154] VM Opanasenko, Sh Kh Fazilov, SS Radjabov, and Sh S Kakharov. 2024. Multilevel Face Recognition System. *Cybernetics and Systems Analysis*, 60, 1, 146–151.

[155] Bengie L Ortiz, Jo Woon Chong, Vibhuti Gupta, Monay Shoushan, Kwanghee Jung, and Tim Dallas. 2022. A Biometric Authentication Technique Using Smartphone Fingertip Photoplethysmography Signals. *IEEE Sensors Journal*, 22, 14, 14237–14249.

[156] Federico Pala and Bir Bhanu. 2017. Iris liveness detection by relative distance comparisons. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 162–169.

[157] Shijia Pan, An Chen, and Pei Zhang. 2013. Securitas: user identification through rgb-nir camera pair on mobile devices. In *Proceedings of the Third ACM workshop on Security and privacy in smartphones & mobile devices*, pp. 99–104.

[158] Omkar Parkhi, Andrea Vedaldi, and Andrew Zisserman. 2015. Deep face recognition. In *BMVC 2015-Proceedings of the British Machine Vision Conference 2015*. British Machine Vision Association.

[159] Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. 2015. Deep Face Recognition. In *Proceedings of the British Machine Vision Conference (BMVC)* Article 41. BMVA Press, (September 2015), pp. 41.1–41.12. ISBN: 1-901725-53-7. DOI: 10.5244/C.29.41.

[160] ASU Pathan, Kamlesh Kumar Thakur, Abhijit Chakraborty, and Mohammad Humayun Kabir. 2019. Fingerprint authentication security: An improved 2-step authentication method with flexibility. *International Journal of Scientific & Engineering Research*, 10, 1.

[161] Annamalai Prakash and Rajeswari Mukesh. 2014. A biometric approach for continuous user authentication by fusing hard and soft traits. *Int. J. Netw. Secur.*, 16, 1, 65–70.

[162] KB Pranav and J Manikandan. 2020. Design and evaluation of a real-time face recognition system using convolutional neural networks. *Procedia Computer Science*, 171, 1651–1659.

[163] Dulyawit Prangchumpol. 2019. Face Recognition for Attendance Management System Using Multiple Sensors. *Journal of Physics: Conference Series*, 1335, 1, (October 2019), 012011. ISSN: 1742-6588,1742-6596. DOI: 10.1088/1742-6596/1335/1/012011.

[164] Michael Preisach. 2022. *System Integrity and Attestation for Biometric Sensors*. Master's thesis. Johannes Kepler University Linz, Institute of Networks and Security, Linz, Austria, (January 2022), 79 pages.

[165] Sen Qiu, Hongkai Zhao, Nan Jiang, Zhelong Wang, Long Liu, Yi An, Hongyu Zhao, Xin Miao, Ruichen Liu, and Giancarlo Fortino. 2022. Multi-sensor information fusion based on machine learning for real applications in human activity recognition: State-of-the-art and research challenges. *Information Fusion*, 80, 241–265.

[166] Enas A Raheem, Sharifah Mumtazah Syed Ahmad, and Wan Azizun Wan Adnan. 2019. Insight on face liveness detection: A systematic literature review. *International Journal of Electrical & Computer Engineering (2088-8708)*, 9, 6.

[167] Emmanuel Ramson, Nehemiah Musa, and John Chaka. 2023. ECG-Based Biometric Schemes for Healthcare: A Systematic Review. 8, (July 2023), 3241–3263. DOI: 10.5281/zenodo.8282855.

[168] Humayan Kabir Rana, Md Shafiul Azam, Mst Rashida Akhtar, Julian MW Quinn, and Mohammad Ali Moni. 2019. A fast iris recognition system through optimum feature extraction. *PeerJ Computer Science*, 5, e184.

[169] Yongming Rao, Jiwen Lu, and Jie Zhou. 2017. Attention-aware deep reinforcement learning for video face recognition. In *Proceedings of the IEEE international conference on computer vision*, pp. 3931–3940.

[170] A Revathi, C Jeyalakshmi, and Karuppusamy Thenmozhi. 2019. Person authentication using speech as a biometric against play back attacks. *Multimedia Tools and Applications*, 78, 2, 1569–1582.

[171] Jacinto Rivero-Hernández, Annette Morales-González, Lester Guerra Denis, and Heydi Méndez-Vázquez. 2021. Ordered Weighted Aggregation Networks for Video Face Recognition. *Pattern Recognition Letters*, 146, 237–243.

[172] Joseph Roth, Xiaoming Liu, Arun Ross, and Dimitris Metaxas. 2013. Biometric authentication via keystroke sound. In *2013 international conference on biometrics (ICB)*. IEEE, pp. 1–8.

[173] Zhang Rui and Zheng Yan. 2018. A survey on biometric authentication: Toward secure and privacy-preserving identification. *IEEE access*, 7, 5994–6009.

[174] Riseul Ryu, Soonja Yeom, Soo-Hyung Kim, and David Herbert. 2021. Continuous multimodal biometric authentication schemes: a systematic review. *IEEE Access*, 9, 34541–34557.

[175] Charles Saavedra, Pamela Smith, and Jessie Peissig. 2013. The relative role of eyes, eyebrows, and eye region in face recognition. *Journal of Vision*, 13, 9, 410–410.

[176] Md Sahidullah, Dennis Alexander Lehmann Thomsen, Rosa Gonzalez Hautamäki, Tomi Kinnunen, Zheng-Hua Tan, Robert Parts, and Martti Pitkänen. 2017. Robust voice liveness detection and speaker verification using throat microphones. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26, 1, 44–56.

[177] Anush Sankaran, Aayush Jain, Tarun Vashisth, Mayank Vatsa, and Richa Singh. 2017. Adaptive latent fingerprint segmentation using feature selection and random decision forest classification. *Information Fusion*, 34, 1–15.

[178] Manisha Sapkale and SM Rajbhoj. 2016. A biometric authentication system based on finger vein recognition. In *2016 International Conference on Inventive Computation Technologies (ICICT)*. Volume 3. IEEE, pp. 1–4.

[179] Neyire Deniz Sarier. 2021. Multimodal biometric authentication for mobile edge computing. *Information Sciences*, 573, (September 2021), 82–99. ISSN: 0020-0255. DOI: 10.1016/j.ins.2021.05.036.

[180] Mohamed Sayed and Faris Baker. 2018. Thermal face authentication with convolutional neural network. *J. Comput. Sci*, 14, 12, 1627–1637.

[181] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823.

[182] Philipp Schwarz, **Hofer, Philipp**, and Josef Scharinger. 2022. Gait Recognition Using 3D View-Transformation Model. In *International Conference on Computer Aided Systems Theory*. Springer, pp. 452–459.

[183] Philipp Schwarz, Josef Scharinger, and **Hofer, Philipp**. 2021. Gait recognition with densePose energy images. In *International Conference on Systems, Signals and Image Processing*. Springer, pp. 65–70.

[184] Sandeep Singh Sengar, U Hariharan, and K Rajkumar. 2020. Multimodal biometric authentication system using deep learning method. In *2020 International Conference on Emerging Smart Computing and Informatics (ESCI)*. IEEE, pp. 309–312.

[185] Sandeep Singh Sengar, U. Hariharan, and K. Rajkumar. 2020. Multimodal Biometric Authentication System using Deep Learning Method. In IEEE, Pune, India, (March 2020), pp. 309–312. ISBN: 978-1-72815-263-9. DOI: 10.1109/ESCI48226.2020.9167512.

[186] Soumyadip Sengupta, Jun-Cheng Chen, Carlos Castillo, Vishal M Patel, Rama Chellappa, and David W Jacobs. 2016. Frontal to profile face verification in the wild. In *2016 IEEE winter conference on applications of computer vision (WACV)*. IEEE, pp. 1–9.

[187] Alireza Sepas-Moghaddam and Ali Etemad. 2022. Deep gait recognition: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 45, 1, 264–284.

[188] Sefik Ilkin Serengil and Alper Ozpinar. 2020. Lightface: A hybrid deep face recognition framework. In *2020 innovations in intelligent systems and applications conference (ASYU)*. IEEE, pp. 1–5.

[189] Tuanjie Shao and Dongkun Shin. 2022. Structured Pruning for Deep Convolutional Neural Networks via Adaptive Sparsity Regularization. In *2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)*. IEEE, pp. 982–987.

[190] Clark D Shaver and John M Acken. 2016. A brief review of speaker recognition technology.

[191] Maneet Singh, Richa Singh, and Arun Ross. 2019. A comprehensive overview of biometric fusion. *Information Fusion*, 52, 187–205.

[192] Manminder Singh and AS Arora. 2018. A novel face liveness detection algorithm with multiple liveness indicators. *Wireless Personal Communications*, 100, 1677–1687.

[193] Ivo Sluganovic, Marc Roeschlin, Kasper B Rasmussen, and Ivan Martinovic. 2018. Analysis of reflexive eye movements for fast replay-resistant biometric authentication. *ACM Transactions on Privacy and Security (TOPS)*, 22, 1, 1–30.

[194] Daniel F Smith, Arnold Wiliem, and Brian C Lovell. 2015. Face recognition on consumer devices: Reflections on replay attacks. *IEEE Transactions on Information Forensics and Security*, 10, 4, 736–745.

[195] Jesús Solano, Christian Lopez, Esteban Rivera, Alejandra Castelblanco, Lizzy Tengana, and Martin Ochoa. 2020. Scrap: synthetically composed replay attacks vs. adversarial machine learning attacks against mouse-based biometric authentication. In *Proceedings of the 13th ACM Workshop on Artificial Intelligence and Security*, pp. 37–47.

[196] Baolin Song, Hao Jiang, Li Zhao, and Chengwei Huang. 2017. A bimodal biometric verification system based on deep learning. In *Proceedings of the International Conference on Video and Image Processing*, pp. 89–93.

[197] Lingxue Song, Dihong Gong, Zhifeng Li, Changsong Liu, and Wei Liu. 2019. Occlusion robust face recognition based on mask learning with pairwise differential siamese network. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 773–782.

[198] Manu Sporny, Dave Longley, and David Chadwick. 2019. Verifiable Credentials Data Model. *W3C Recommendation*. https://www.w3.org/TR/vc-data-model/.

[199] David B Stewart. 2001. Measuring execution time and real-time performance. In *Embedded Systems Conference (ESC)*. Volume 141. Citeseer.

[200] Ahmed Raad Al-Sudani, Shang Gao, Sheng Wen, and Muhmmad Al-Khiza'ay. 2018. Checking an authentication of person depends on RFID with thermal image. In *Security, Privacy, and Anonymity in Computation, Communication, and Storage: 11th International Conference and Satellite Workshops, SpaCCS 2018, Melbourne, NSW, Australia, December 11-13, 2018, Proceedings 11*. Springer, pp. 371–380.

[201] Bharath Sudharsan, Peter Corcoran, and Muhammad Intizar Ali. 2022. Smart speaker design and implementation with biometric authentication and advanced voice interaction capability. *arXiv preprint arXiv:2207.10811*.

[202] Ziwen Sun, Yao Wang, Gang Qu, and Zhiping Zhou. 2016. A 3-D hand gesture signature based biometric authentication system for smartphones. *Security and Communication Networks*, 9, 11, 1359–1373.

[203] Aditya Sundararajan, Arif I Sarwat, and Alexander Pons. 2019. A survey on modality characteristics, performance evaluation metrics, and security for traditional and wearable biometric systems. *ACM Computing Surveys (CSUR)*, 52, 2, 1–36.

[204] Dhiraj Sunehra. 2014. Fingerprint based biometric ATM authentication system. *International journal of engineering inventions*, 3, 11, 22–28.

[205] Rahmad Syalevi, Aji Prasetyo, and Rizal Fathoni Aji. 2024. Study on the Implementation of Multimodal Continuous Authentication in Smartphones: A Systematic Review. *International Journal of Advanced Computer Science & Applications*, 15, 2.

[206] Rafat Jamal Tazim, Md Messal Monem Miah, Sanzida Sayedul Surma, Mohammad Tariqul Islam, Celia Shahnaz, and Shaikh Anowarul Fattah. 2018. Biometric authentication using CNN features of dorsal vein pattern extracted from NIR image. In *TENCON 2018-2018 IEEE Region 10 Conference*. IEEE, pp. 1923–1927.

[207] Fatemeh Tehranipoor, Nima Karimian, Paul A Wortman, and John A Chandy. 2018. Low-cost authentication paradigm for consumer electronics within the internet of wearable fitness tracking applications. In *2018 IEEE international conference on consumer electronics (ICCE)*. IEEE, pp. 1–6.

[208] Shejin Thavalengal and Peter Corcoran. 2016. User authentication on smartphones: Focusing on iris biometrics. *IEEE Consumer Electronics Magazine*, 5, 2, 87–93.

[209] The Tor Project. 2023. Onion Services. Accessed: August 14, 2024. (2023). Retrieved 08/14/2024 from https://community.torproject.org/onion-services/.

[210] AS Tolba, AH El-Baz, and AA El-Harby. 2006. Face recognition: A literature review. *International Journal of Signal Processing*, 2, 2, 88–103.

[211] Tsung-Han Tsai and Shih-An Huang. 2022. Refined U-net: A new semantic technique on hand segmentation. *Neurocomputing*, 495, 1–10.

[212] uidai.gov.in. 2023. Unique Identification Authority of India. Accessed February 2, 2023. https://uidai.gov.in/en/. (2023).

[213] Buhari Ugbede Umar, Olayemi Mikail Olaniyi, Abisoye Blessing Olatunde, Ademoh Agbogunde Isah, Arifa Khatoon Haq, and Isaac Taiye Ajayi. 2022. A bi-factor biometric authentication system for secure electronic voting system. In *2022 IEEE Nigeria 4th International Conference on Disruptive Technologies for Sustainable Development (NIGERCON)*. IEEE, pp. 1–5.

[214] Anthony Ngozichukwuka Uwaechia and Dzati Athiar Ramli. 2021. A comprehensive survey on ECG signals as new biometric modality for human authentication: Recent advances and future challenges. *IEEE Access*, 9, 97760–97802.

[215] Gooljar Veerajay, S Ramiah, and H Vasudavan. 2019. Biometric Bus Ticketing System In Mauritius. *International Journal of Scientific and Technology Research*, 8, 12, 568–571.

[216] Sudip Vhaduri and Christian Poellabauer. 2019. Multi-modal biometric-based implicit authentication of wearable device users. *IEEE Transactions on Information Forensics and Security*, 14, 12, 3116–3125.

[217] P. Viola and M. Jones. 2001. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*. Volume 1, pp. I–I. DOI: 10.1109/CVPR.2001.990517.

[218] Changsheng Wan, Li Wang, and Vir V Phoha. 2018. A survey on gait recognition. *ACM Computing Surveys (CSUR)*, 51, 5, 1–35.

[219] Guo Chun Wan, Meng Meng Li, He Xu, Wen Hao Kang, Jin Wen Rui, and Mei Song Tong. 2020. XFinger-net: pixel-wise segmentation method for partially defective fingerprint based on attention gates and U-net. *Sensors*, 20, 16, 4473.

[220] Caiyong Wang, Jawad Muhammad, Yunlong Wang, Zhaofeng He, and Zhenan Sun. 2020. Towards complete and accurate iris segmentation using deep multi-task attention network for non-cooperative iris recognition. *IEEE Transactions on information forensics and security*, 15, 2944–2959.

[221] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. 2018. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5265–5274.

[222] Qi Wang, Xiangyue Meng, Ting Sun, and Xiangde Zhang. 2022. A light iris segmentation network. *The Visual Computer*, 38, 7, 2591–2601.

[223] Yao Wang, Wandong Cai, Tao Gu, Wei Shao, Yannan Li, and Yong Yu. 2019. Secure your voice: An oral airflow-based continuous liveness detection for voice assistants. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies*, 3, 4, 1–28.

[224] Zhongyuan Wang, Baojin Huang, Guangcheng Wang, Peng Yi, and Kui Jiang. 2023. Masked face recognition dataset and application. *IEEE Transactions on Biometrics, Behavior, and Identity Science*.

[225] Sethapong Wong-In and Paniti Netinant. 2017. Revised software model design for biometric examiner personal verification system. In *Proceedings of the 2017 International Conference on Information Technology*, pp. 237–242.

[226] Libing Wu, Jingxiao Yang, Man Zhou, Yanjiao Chen, and Qian Wang. 2019. LVID: A multimodal biometrics authentication system on smartphones. *IEEE Transactions on Information Forensics and Security*, 15, 1572–1585.

[227] Wei Wu, Yuan Zhang, Yunpeng Li, and Chuanyang Li. 2024. Fusion recognition of palmprint and palm vein based on modal correlation. *Mathematical Biosciences and Engineering*, 21, 2, 3129–3145.

[228] Zhi Wu, Dongheng Zhang, Chunyang Xie, Cong Yu, Jinbo Chen, Yang Hu, and Yan Chen. 2022. RFMask: A simple baseline for human silhouette segmentation with radio signals. *IEEE Transactions on Multimedia*.

[229] David Yambay, Benedict Becker, Naman Kohli, Daksha Yadav, Adam Czajka, Kevin W Bowyer, Stephanie Schuckers, Richa Singh, Mayank Vatsa, Afzel Noore, et al. 2017. LivDet iris 2017—Iris liveness detection competition 2017. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, pp. 733–741.

[230] Jiaolong Yang, Peiran Ren, Dongqing Zhang, Dong Chen, Fang Wen, Hongdong Li, and Gang Hua. 2017. Neural aggregation network for video face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4362–4371.

[231] Qing Yang, Jiachen Mao, Zuoguan Wang, and Li Hai. 2021. Dynamic Regularization on Activation Sparsity for Neural Network Efficiency Improvement. *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, 17, 4, 1–16.

[232] Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang. 2016. WIDER FACE: A Face Detection Benchmark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5525–5533. DOI: 10.1109/CVPR.2016.596.

[233] Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang. 2016. Wider face: A face detection benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5525–5533.

[234] Masaki Yasuhara, Isao Nambu, and Shohei Yano. 2022. Bilateral ear acoustic authentication: A biometric authentication system using both ears and a special earphone. *Applied Sciences*, 12, 6, 3167.

[235] Chengsheng Yuan, Xinting Li, QM Jonathan Wu, Jin Li, and Xingming Sun. 2017. Fingerprint liveness detection from different fingerprint materials using convolutional neural network and principal component analysis. *Computers, Materials & Continua*, 53, 3, 357–371.

[236] Chengsheng Yuan, Xingming Sun, and Rui Lv. 2016. Fingerprint liveness detection based on multi-scale LPQ and PCA. *China Communications*, 13, 7, 60–65.

[237] Nuhu Yusuf, Kamalu Abdullahi Marafa, Kamila Ladan Shehu, Hussaini Mamman, and Mustapha Maidawa. 2020. A survey of biometric approaches of authentication. *International Journal of Advanced Computer Research*, 10, 47, 96–104.

[238] Stefanos Zafeiriou, Cha Zhang, and Zhengyou Zhang. 2015. A survey on face detection in the wild: past, present and future. *Computer Vision and Image Understanding*, 138, 1–24.

[239] Mehrzad Zargarzadeh and Keivan Maghooli. 2013. A behavioral biometric authentication system based on memory game. *Biosci. Biotechnol. Res. Asia*, 10, 2, 781–787.

[240] Ye Zhan, Aditya Singh Rathore, Giovanni Milione, Yuehang Wang, Wenhan Zheng, Wenyao Xu, and Jun Xia. 2020. 3D finger vein biometric authentication with photoacoustic tomography. *Applied Optics*, 59, 28, 8751–8758.

[241] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. 2016. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23, 10, 1499–1503.

[242] Linghan Zhang, Sheng Tan, Zi Wang, Yili Ren, Zhi Wang, and Jie Yang. 2020. Viblive: A continuous liveness detection for secure voice user interface in iot environment. In *Proceedings of the 36th Annual Computer Security Applications Conference*, pp. 884–896.

[243] Linghan Zhang, Sheng Tan, and Jie Yang. 2017. Hearing your voice is not enough: An articulatory gesture based liveness detection for voice authentication. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 57–71.

[244] Linghan Zhang, Sheng Tan, Jie Yang, and Yingying Chen. 2016. Voicelive: A phoneme localization based liveness detection for voice authentication on smartphones. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1080–1091.

[245] Shifeng Zhang, Xiangyu Zhu, Zhen Lei, Hailin Shi, Xiaobo Wang, and Stan Z Li. 2017. Faceboxes: A CPU real-time face detector with high accuracy. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, pp. 1–9.

[246] Xiang Zhang, Lina Yao, Chaoran Huang, Tao Gu, Zheng Yang, and Yunhao Liu. 2020. DeepKey: a multimodal biometric authentication system via deep decoding gaits and brainwaves. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11, 4, 1–24.

[247] Xinman Zhang, Dongxu Cheng, Pukun Jia, Yixuan Dai, and Xuebin Xu. 2020. An efficient android-based multimodal biometric authentication system with face and voice. *IEEE Access*, 8, 102757–102772.

[248] Zhishuai Zhang, Wei Shen, Siyuan Qiao, Yan Wang, Bo Wang, and Alan Yuille. 2020. Robust face detection via learning small faces on hard images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1361–1370.

[249] Shiyu Zhao, Wankou Yang, and Yangang Wang. 2018. A new hand segmentation method based on fully convolutional network. In *2018 Chinese Control And Decision Conference (CCDC)*. IEEE, pp. 5966–5970.

[250] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu. 2019. Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, 30, 11, 3212–3232.

[251] Jingxiao Zheng, Rajeev Ranjan, Ching-Hui Chen, Jun-Cheng Chen, Carlos D Castillo, and Rama Chellappa. 2020. An automatic system for unconstrained video-based face recognition. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2, 3, 194–209.

[252] Tianyue Zheng and Weihong Deng. 2018. Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments. *Beijing University of Posts and Telecommunications, Tech. Rep*, 5, 7.

[253] Dexing Zhong, Xuefeng Du, and Kuncai Zhong. 2019. Decade progress of palmprint recognition: A brief survey. *Neurocomputing*, 328, 16–28.

Appendix A

Code

To avoid adding unnecessary pages to this thesis, I have included the complete source code used in this thesis as a file attachment, integrated into the PDF document itself. This integration was achieved using the attachfile2 LaTeX package[1], which allows for the embedding of arbitrary files directly within the PDF structure, which is possible since PDF 1.3. The attached ZIP file containing the source code can be accessed either by clicking the embedded link within this document, if supported by your PDF viewer (📎), or by using PDF manipulation tools such as the `pdfdetach` command.

This approach offers useful benefits for the long-term preservation and accessibility of research materials. By embedding the relevant code within the PDF, the implementation details stay closely linked to the thesis text, which makes it easier for others to reference and reproduce the results in the future. Additionally, this self-contained format helps reduce the risk of the code becoming separated from its documentation.

[1] https://ctan.org/pkg/attachfile2